

ESTE EXEMPLAR CORRESPONDE A REDAÇÃO FINAL DA
TESE DEFENDIDA POR Jaime Armando
Delgado Vargas E APROVADA
PELA COMISSÃO JULGADORA EM 15 / 02 / 2012


.....
ORIENTADOR

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

Jaime Armando Delgado Vargas

**Localização E Navegação De Robô Autônomo
Através De Odometria E Visão Estereoscópica**

Campinas, 2012.

Jaime Armando Delgado Vargas

Localização E Navegação De Robô Autônomo Através De Odometria E Visão Estereoscópica

Dissertação apresentada ao Curso de
Mestrado da Faculdade de Engenharia
Mecânica da Universidade Estadual de
Campinas, como requisito para a obtenção do
título de Mestre em Engenharia Mecânica.

Área de Concentração: Mecânica dos
sólidos e projeto mecânico

Orientador:
Prof. Dr. Paulo Roberto Gardel Kurka

Campinas, 2012
S.P. - Brasil

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

V426L Vargas, Jaime Armando Delgado
Localização e navegação de robô autônomo através
de odometria e visão estereoscópica / Jaime Armando
Delgado Vargas . --Campinas, SP: [s.n.], 2012.

Orientador: Paulo Roberto Gardel Kurka.
Dissertação de Mestrado - Universidade Estadual de
Campinas, Faculdade de Engenharia Mecânica.

1. Robôs móveis. 2. Visão por computador. 3.
Navegação de robôs móveis. I. Kurka, Gardel Paulo
Roberto. II. Universidade Estadual de Campinas.
Faculdade de Engenharia Mecânica. III. Título.

Título em Inglês: Localization and navigation of an autonomous mobile robot
through odometry and stereoscopic vision

Palavras-chave em Inglês: Mobile robot, Computer vision, Mobile robot
navigation

Área de concentração: Mecânica dos Sólidos e Projeto Mecânico

Titulação: Mestre em Engenharia Mecânica

Banca examinadora: Eleri Cardozo, Niederauer Mastelari

Data da defesa: 15-02-12

Programa de Pós Graduação: Engenharia Mecânica

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO (DPM)**

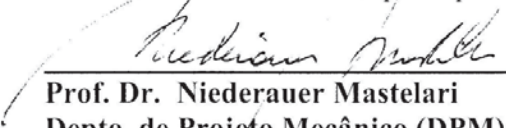
DISSERTAÇÃO DE MESTRADO ACADÊMICO

**Localização e navegação de robô autônomo
através de odometria e visão estereoscópica**

Autor: Jaime Armando Delgado Vargas

Orientador: Prof. Dr. Paulo Roberto Gardel Kurka

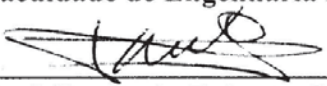
A Banca Examinadora composta pelos membros abaixo aprovou esta Dissertação:



Prof. Dr. Niederauer Mastelari
Depto. de Projeto Mecânico (DPM)
Faculdade de Engenharia Mecânica - UNICAMP



Prof. Dr. Eleri Cardozo
Depto. de Engenharia da Computação e Automação Industrial (DCA)
Faculdade de Engenharia Elétrica - UNICAMP



Prof. Dr. Paulo Roberto Gardel Kurka
Depto. de Projeto Mecânico (DPM)
Faculdade de Engenharia Mecânica - UNICAMP

Campinas, 15 de fevereiro de 2012.

Dedico este trabalho à minha família.

Agradecimentos

Este trabalho não poderia ser terminado sem a ajuda de diversas pessoas e instituições. Primeiramente agradeço a Deus e aos meus pais Jaime e Ana Rosa pelas ferramentas e valores que me ensinaram. A meu irmão Daniel que é meu colega de caminhada. A minha namorada Paula por me acompanhar e apoiar todos os dias.

Aos professores da Colômbia, Oscar Aviles e Dario Amaya por me iniciar no mundo da pesquisa. Ao professor Néstor Gabriel Sepúlveda Q., Ph.D que levou o conhecimento em mecatrônica para a Colômbia, ferramenta fundamental na minha formação.

Ao meu orientador, quem me acompanhou, guiou e ajudou no desenvolvimento deste trabalho. Ao professor Eleri, quem forneceu os materiais e conhecimentos em robótica e junto com o professor Niederauer examinaram este trabalho.

A todos os professores e colegas do departamento, que ajudaram de forma direta e indireta na conclusão deste trabalho.

Agradeço a CNPq pelo apoio, através da bolsa de mestrado. A capes e FAPESP pelo o apoio ao laboratório e departamento.

No te des por vencido, ni aun vencido,
no te sientas esclavo, ni aun esclavo;
trémulo de pavor, piénsate bravo,
y arremete feroz, ya mal herido.
Ten el tesón del clavo enmohecido
que ya viejo y ruin, vuelve a ser clavo;
no la cobarde estupidez del pavo
que amaina su plumaje al primer ruido.
Procede como Dios que nunca llora;
o como Lucifer, que nunca reza;
o como el robledal, cuya grandeza
necesita del agua y no la implora...
Que muerda y vocifere vengadora,
ya rodando en el polvo, tu cabeza!
¡PIU AVANTI!

Resumo

Este trabalho apresenta a implementação de um sistema de navegação com visão estereoscópica em um robô móvel, que permite a construção de mapa de ambiente e localização. Para isto é necessário conhecer o modelo cinemático do robô, técnicas de controle, algoritmos de identificação de características em imagens (*features*), reconstrução 3D com visão estereoscópica e algoritmos de navegação. Utilizam-se métodos para a calibração de câmera desenvolvida no âmbito do grupo de pesquisa da FEM/UNICAMP e da literatura. Resultados de análises experimentais e teóricas são comparados. Resultados adicionais mostram a validação do algoritmo de calibração de câmera, acurácia dos sensores, resposta do sistema de controle, e reconstrução 3D. Os resultados deste trabalho são de importância para futuros estudos de navegação robótica e calibração de câmeras.

Palavras Chave: Robô Móvel; Visão Estereoscópica, Odometria visual.

Abstract

This paper presents a navigation system with stereoscopic vision on a mobile robot, which allows the construction of environment map and location. In that way must know the kinematic model of the robot, algorithms for identifying features in images (features) as a Sift, 3D reconstruction with stereoscopic vision and navigation algorithms. Methods are used to calibrate the camera developed within the research group of the FEM / UNICAMP and literature. Results of experimental and theoretical analyzes are compared. Additional results show the validation of the algorithm for camera calibration, accuracy of sensors, control system response, and 3D reconstruction. These results are important for future studies of robotic navigation and calibration of cameras.

Key Words: Mobile Robot; Stereoscopic Vision, Visual Odometry.

Lista de Ilustrações

FIGURA 1-1 IDENTIFICAÇÃO DE CARACTERÍSTICAS NO MUNDO, SISTEMA DE VISÃO ESTEREOSCÓPICA ATIVO (DAVISON, ET AL., 2002)	2
FIGURA 1-2 ESTIMAÇÃO 3D DE MÚLTIPLOS OBJETOS (ZHU, ET AL., 2004)	3
FIGURA 1-3 SISTEMA DE VISÃO ESTEREOSCÓPICA COM DOIS ROBÔS (ZHU, ET AL., 2004)	3
FIGURA 1-4 RESULTADOS EXPERIMENTAIS DO TRABALHO (SAEEDI, ET AL., 2006), AMBIENTE (OUTDOOR), NÃO ESTRUTURADOS.	4
FIGURA 1-5 VEÍCULO VOADOR NÃO TRIPULADO <i>UNMANNED AERIAL VEHICLE</i> (UAV) (STEDER, ET AL., 2008)	4
FIGURA 1-6 MAPA DO MUNDO GERADO POR VISÃO, CÂMERA ESTEREOSCÓPICAS USADAS (STEDER, ET AL., 2008)	5
FIGURA 1-7 EXPERIMENTO DE NAVEGAÇÃO SLAM EM AMBIENTES INDOOR (STEDER, ET AL., 2008)	5
FIGURA 1-8 CÂMERA ESTEREOSCÓPICA DE SEIS GRAUS DE LIBERDADE (PAZ, ET AL., 2008)	6
FIGURA 1-9 INCERTEZA DA VISÃO ESTÉREA PARA DIFERENTES LONGITUDES (PAZ, ET AL., 2008)	6
FIGURA 1-10 SUBMARINO SIRIUS, SISTEMAS DE VISÃO ESTEREOSCÓPICA (MAHON, ET AL., 2008)	7
FIGURA 1-11 MAPA RECONSTRUÍDO POR DR E POR VISÃO ESTEREOSCÓPICA, IMAGENS ESTÉREO EM DOIS TEMPOS (MAHON, ET AL., 2008)	7
FIGURA 1-12 RECONSTRUÇÃO 3D DA TRAJETORIA E FOTOGRAFIA DO AMBIENTE REAL (SILVEIRA, ET AL., 2007)	8
FIGURA 1-13 SISTEMA DE COORDENADAS DO ROBÔ EM RELAÇÃO A UM OBJETIVO	10
FIGURA 1-14 SISTEMA DE COORDENADAS DO ROBÔ EM COORDENADAS POLARES	11
FIGURA 2-1 PARÂMETROS USADOS NA CALIBRAÇÃO DA CÂMERA COM MATLAB®, (A) APRESENTAÇÃO DO CONJUNTO DE IMAGENS DE CALIBRAÇÃO, (B) MÉTODO DE CALIBRAÇÃO DE CÂMERA DESENVOLVIDO NO GRUPO DE PESQUISA	15
FIGURA 2-2 PARÂMETROS USADOS NA CALIBRAÇÃO DA CÂMERA COM MATLAB®	15
FIGURA 2-3 MÉTODO DE CALIBRAÇÃO DE CÂMERA DESENVOLVIDO NO GRUPO DE PESQUISA	16
FIGURA 2-4 CONVOLUÇÃO ENTRE AS IMAGENS DISTORCIDAS L (G) GERA A IMAGEM DE DIFERENÇAS GAUSSIANAS D (DOG) EM CADA OITAVA, (LOWE, 2004).	17
FIGURA 2-5 VIZINHANÇA 3D DE MOORE DO <i>KEYPOINT</i> COM RADIO 1 (LOWE, 2004).	18
FIGURA 2-6 RESULTADOS EXPERIMENTAIS DO ALGORITMO SIFT	19
FIGURA 2-7 GEOMETRIA EPIPOLAR DE DUAS CÂMERAS, VISTAS FRONTAL E TRASEIRAS.	20
FIGURA 2-8 PROJEÇÃO DE UM PONTO EM UM SISTEMA ESTEREOSCÓPICO	21
FIGURA 2-9 SISTEMA DE VISÃO ESTEREOSCÓPICA COM CÂMERAS PARALELAS.	23
FIGURA 2-10 CORRESPONDÊNCIA NAS IMAGENS	24
FIGURA 3-1 SISTEMA DE COORDENADAS DO ROBÔ E DAS CÂMERAS	25
FIGURA 3-2 VETORES DA CÂMERA DIREITA EM DOIS INSTANTES DE TEMPO.	26
FIGURA 3-3 DESLOCAMENTO DO ROBÔ	26
FIGURA 3-4 ODOMETRIA DE SISTEMAS DE COORDENADAS CONSECUTIVAS	27
FIGURA 4-1 ROBÔ MÓBILE PIONNER 3D-X	29
FIGURA 4-2 ARQUITETURA DO ROBÔ MÓBILE PIONNER 3D-X	29

FIGURA 4-3 MODELO DA CÂMERA ESTEREOSCÓPICA	30
FIGURA 4-4 MODELO 3D DO ROBÔ PIONEER	30
FIGURA 4-5 AMBIENTE 3D DE CALIBRAÇÃO.....	31
FIGURA 4-6 IMAGENS DA CAIXA DE CALIBRAÇÃO	31
FIGURA 4-7 TEXTURAS DO MODELO, MODELO DO CORREDOR DO LABORATÓRIO.....	32
FIGURA 4-8 CÂMERA WEBCAM LOGITECH C160	32
FIGURA 4-9 CÂMERA ESTEREOSCÓPICA	32
FIGURA 4-10 ÂNGULO DE VISTA DA CÂMERA ESTEREOSCÓPICA	33
FIGURA 5-1 RECONSTRUÇÃO 3D DAS GRILAS E A LOCAÇÃO DAS CÂMERAS NO ESPAÇO.	34
FIGURA 5-2 IMAGENS ESTEREOSCÓPICAS PARA TESTAR A CALIBRAÇÃO	35
FIGURA 5-3 DISTRIBUIÇÃO NORMAL DA CÂMERA ESTEREOSCÓPICA	36
FIGURA 5-4 RECONSTRUÇÃO 3D DO AMBIENTE	37
FIGURA 5-5 REPRESENTAÇÃO DOS SISTEMAS DE COORDENADAS.....	38
FIGURA 5-6 REPRESENTAÇÃO DOS PONTOS DESCARTADOS PELO FILTRO	39
FIGURA 5-7 IMAGENS DA SIMULAÇÃO DO AMBIENTE VIRTUAL.....	39
FIGURA 5-8 CORRELAÇÕES DA IMAGEM DA DIREITA NO AMBIENTE VIRTUAL.....	40
FIGURA 5-9 TRAJETÓRIA RECONSTRUÍDA DA SIMULAÇÃO NO AMBIENTE VIRTUAL.....	40
FIGURA 5-10 DESLOCAMENTOS NO EIXO Z, AMBIENTE REAL.....	41
FIGURA 5-11 IMAGENS DA CÂMERA ESQUERDA NO DESLOCAMENTO LINEAR.....	41
FIGURA 5-12 RECONSTRUÇÃO DA ODOMETRIA VISUAL E DOS ENCODERS, TRAJETO LINHA 1D, FILTRO 30%	42
FIGURA 5-13 ERROS LOCAIS DA TRAJETÓRIA, LINHA 1D, COM O FILTRO EM 30%.....	42
FIGURA 5-14 ERRO E NÚMERO DE PONTOS DA TRAJETÓRIA,	43
FIGURA 5-15 RECONSTRUÇÃO DA ODOMETRIA VISUAL E DOS ENCODERS, TRAJETO LINHA 1D, FILTRO 70%.	43
FIGURA 5-16 ERROS LOCAIS DA TRAJETÓRIA, LINHA 1D, COM O FILTRO EM 70%.....	44
FIGURA 5-17 ERRO E NÚMERO DE PONTOS DA TRAJETÓRIA,	44
FIGURA 5-18 IMAGENS DA NAVEGAÇÃO NO TRAJETO SENOIDAL	45
FIGURA 5-19 RECONSTRUÇÃO DA ODOMETRIA VISUAL E DOS ENCODERS, TRAJETO SENOIDAL 2D, FILTRO 30%.	46
FIGURA 5-20 ERROS LOCAIS DA TRAJETÓRIA, SENOIDAL 2D, COM O FILTRO EM 30%	47
FIGURA 5-21 61 ERRO E NÚMERO DE PONTOS DA TRAJETÓRIA,	47
FIGURA 5-22 CORRELAÇÕES REPRESENTADAS COMO VETORES EM COORDENADAS ESFÉRICAS.....	48
FIGURA 5-23 CORRELAÇÕES REPRESENTADAS COMO VETORES SOBRE A SOMATÓRIA DAS IMAGENS FILTRO 30%.....	48
FIGURA 5-24 RECONSTRUÇÃO DE UMA TRAJETÓRIA EM FORMATO DE SENOIDAL 2D, COM O FILTRO EM 70%.....	49
FIGURA 5-25 CORRELAÇÕES REPRESENTADAS COMO VETORES SOBRE A SOMATÓRIA DAS IMAGENS FILTRO 70%.....	50
FIGURA 5-26 ERROS LOCAIS DA TRAJETÓRIA, SENOIDAL 2D, COM O FILTRO EM 70%	50
FIGURA 5-27 ERRO E NUMERO DE PONTOS DA TRAJETÓRIA,	51

Lista de Abreviaturas e Siglas

Abreviações

SLAM – Localização e Mapeamento Simultâneos

vSLAM – Visual Localização e Mapeamento Simultâneos

ViPR – Reconhecimento de Padrões Visuais em inglês “Visual Pattern Recognition”

ccd – Sensor eletrônico de uma câmera, em inglês *charge-coupled device*

Nomenclatura

O_l : Centro geométrico esquerdo da câmera estereoscópica

O_r : Centro geométrico direito da câmera estereoscópica

e_l : Epipolo da câmera esquerda

e_r : Epipolo da câmera direita

l_l : Linha epipolar da câmera esquerda

l_r : Linha epipolar da câmera direita

P : Ponto no espaço

P_k : Vetor de pontos no espaço

p_l : Projeção do ponto P na imagem da câmera direita

p_r : Projeção do ponto P na imagem da câmera esquerda

c_l : Centro da imagem esquerda

c_r : Centro da imagem direita

bl : Linha base, distância que separa os centros das câmeras

M: Matriz dos parâmetros da câmera (intrínsecos e extrínsecos)

F: Matriz fundamental

T : Vetor de translação

R: Matriz de rotação

x_l : Distância entre p_l e c_l
 x_r : Distância entre p_r e c_r
 d : Disparidade
 x, y, z : Coordenadas do ponto P no sistema global de referências
 z_{lk} : Profundidade do ponto P_k , referenciado no sistema de coordenadas esquerdo.
 z_{rk} : Profundidade do ponto P_k , referenciado no sistema de coordenadas direito.
 f_l : Parâmetro intrínseco, distância focal da câmera esquerda.
 f_r : Parâmetro intrínseco, distância focal da câmera direita.
 \mathbf{T}_{mi} : Vetor de translação do robô
 \mathbf{R}_{mi} : Matriz de rotação do robô
 \mathbf{P}_i : Matriz de vetores de deslocamento no tempo atual
 \mathbf{P}_{i-1} : Matriz de vetores de deslocamento no tempo apriori
 \mathbf{X}_i : Posição do robô, com relação á informação de odometria
 x_R, y_R : Coordenadas do robô no espaço
 θ : Orientação do robô em graus
 ρ : Distância do centro do robô até o alvo
 β : Orientação do alvo em graus
 α : Erro de orientação
 ω : Velocidade angular do robô Pioneer
 v : Velocidade de translação do robô

SUMÁRIO

Resumo.....	vii
Abstract.....	viii
Lista de Ilustrações.....	ix
Lista de Abreviaturas e Siglas	xi
SUMÁRIO..	xiii
1 INTRODUÇÃO	1
1.1 Revisão da Literatura	1
1.2 Cinemática Do Robô Móvel.....	9
1.2.1 Erro de odometria.....	9
1.2.2 Modelo Cinemático	9
1.3 Objetivos	12
1.4 Organização do texto.....	13
2 ESTEREOSCOPIA	14
2.1 Calibração da câmera	14
2.2 Identificação de características.....	16
2.2.1 Detecção dos extremos de escala e espaço	16
2.2.2 Localização do keypoint.....	18
2.2.3 Orientação do keypoint	18
2.3 Visão estereoscópica	20
2.3.1 Geometria epipolar	20
2.3.2 Reconstrução 3D	21
2.3.3 Disparidade.....	22
2.3.4 Correspondência.....	24
3 LOCALIZAÇÃO PELA ODOMETRIA VISUAL.....	25
4 DISPOSITIVOS USADOS NA IMPLEMENTAÇÃO DOS MÉTODOS DE LOCALIZAÇÃO	28

4.1	Robô Móvel Pioneer 3D-X	28
4.2	Câmera estereoscópica	29
4.2.1	Simulação da câmera estereoscópica com 3DsMax.....	29
4.2.2	Construção da câmera estereoscópica	32
5	RESULTADOS	34
5.1	Calibração estereoscópica utilizando o algoritmo de Bouguet	34
5.2	Reconstrução do ambiente	36
5.3	Odometria visual	37
5.3.1	Filtro de correlações	38
5.3.2	Trajectoria Linear.....	39
5.3.3	Trajeto senoidal	45
6	DISCUSSÕES, CONCLUSÕES.....	52
Referências A		
ANEXOS A - Produção Bibliográfica		F
ANEXOS B - Diagrama de fluxo do algoritmo de odometria.....		G

1 INTRODUÇÃO

Robôs móveis e os manipuladores ARMS (*Axis Articulated Robots*) são duas importantes aplicações no universo de automação e robótica. A maior diferença encontra-se na capacidade dos robôs móveis de deslocar-se no ambiente, ou seja, não estão em uma localização fixa. Esta habilidade tem sido objeto de estudos, fomentando o conhecimento em diferentes áreas de pesquisa. Uma delas é a representação da posição correta de robôs. Para isto, é preciso gerar um mapeamento do ambiente de movimentação do robô (mundo), ao mesmo tempo em que esse se desloca. Tal técnica é conhecida como SLAM (*Simultaneous localization and mapping*). Um dos primeiros trabalhos a empregar esta metodologia, publicado no ano de 1990 (R. Smith, 1990), apresenta a incerteza associada às estimativas de cálculo da posição espacial de um robô, e desde então se tem aplicado e desenvolvido essa tecnologia. Por outro lado, a visão é um meio que fornece um grande volume de informações sensoriais aos seres humanos. Isso vem despertando o interesse dos pesquisadores, que tem desenvolvido para esse fim, diferentes tipos de câmeras e algoritmos de processamento de informações, como a extração de características e reconstrução 3D do ambiente, entre outros. Neste trabalho usa-se visão estereoscópica, para resolver o problema de localização de robôs móveis (vSLAM), obtendo pontos de interesse em duas imagens e correlacionando-os, com a finalidade de gerar informações de localização e estimativas do caminho percorrido.

1.1 Revisão da Literatura

Nas aplicações de robótica, como na automação industrial e movimentação autônoma, é necessário representar a pose (posição e orientação) correta do robô. Um dos primeiros trabalhos que utiliza a metodologia SLAM (R. Smith, 1990), apresenta a incerteza da estimativa da posição espacial do robô, justificando a necessidade de se implementar algoritmos de visão computacional para a localização dos robôs. Duas são as técnicas de visão computacional que podem atender as tarefas de localização: visão estereoscópica no espaço Euclidiano [2-4,6,7] e visão monocular no espaço projetivo (Silveira, et al., 2008), (Silveira, et al., 2007). Além disso, os algoritmos são classificados tendo em conta o ambiente de trabalho, ou seja, se o robô vai

trabalhar em locais fechados como corredores, quartos ou escritórios (chamados ambientes *indoors*) ou em lugares abertos onde as distâncias a os objetos são maiores (ambientes *outdoors*). Apresenta-se a seguir, para efeito de situação do problema, uma síntese de diferentes projetos de pesquisa que buscam uma solução para o problema SLAM com visão estereoscópica.

No departamento de "Engineering Science", na universidade de Oxford, foi feita a medição de um campo de vista (SLAM) com o uso de visão e detenção automática de características. O trabalho apresenta um modelo de localização autônoma utilizando visão estereoscópica ativa de alto rendimento, gerando uma estimativa da posição do robô e fazendo a manutenção automática do mapa, conforme apresentado na Figura 1-1. As características do ambiente são identificadas com um algoritmo dedicado de detecção de bordas. O sistema de visão estereoscópica ativo consiste em duas câmeras com motores, cada uma com dois graus de liberdade. A intervenção de motores nas câmeras faz que o sistema se comporte de uma forma dinâmica, criando um sistema ativo (Davison, et al., 2002).



Figura 1-1 Identificação de características no mundo, sistema de visão estereoscópica ativo (Davison, et al., 2002)

Na universidade "Massachusetts Amherst" foi desenvolvido um sistema de visão estereoscópica panorâmica adaptativa, para seguimento e localização de pessoas. Esse sistema é composto por robôs cooperativos, cada um possui um sistema de visão monocular. O sistema estereoscópico gerado realiza seguimento (*tracking*) e detecta movimentação humana em ambientes fechados (*indoors*). O sistema é apresentado na Figura 1-2, onde são identificadas duas pessoas num ambiente 3D.

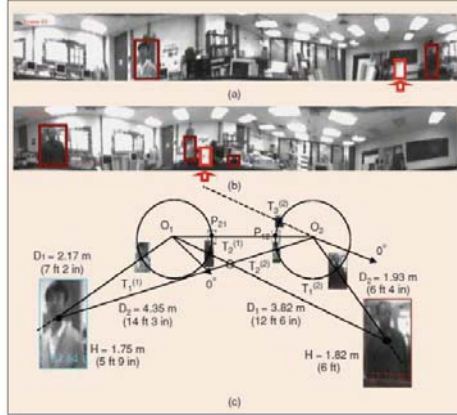


Figura 1-2 Estimação 3D de múltiplos objetos (Zhu, et al., 2004)

A calibração é feita direcionando uma câmera para a outra, o que elimina a necessidade do uso de objetos padrões de calibração no ambiente (Zhu, et al., 2004). Os robôs são apresentados abaixo na Figura 1-3

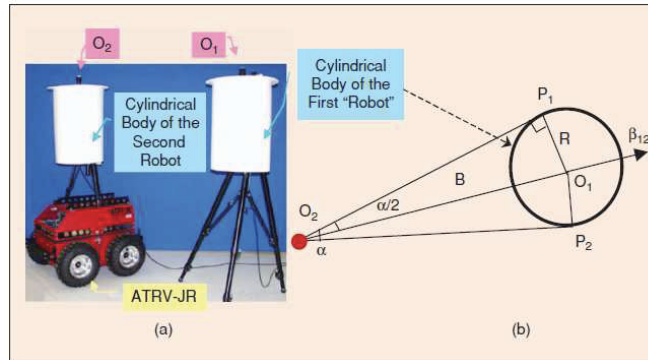


Figura 1-3 Sistema de visão estereoscópica com dois robôs (Zhu, et al., 2004)

Um sistema de visão estereoscópica em ambientes abertos (*outdoors*) foi desenvolvido pelo “Natural Sciences and Engineering Research Council of Canada” com colaboração do “Canadian IRIS/PRECARN Network of Centers of Excellence”, que apresenta a localização de um robô num ambiente externo, sem referências (*land marks*). O robô é dotado de três câmeras monoculares. A pose do robô (localização em relação a um sistema de referências) é estimada, e é feita uma aproximação da propagação do erro. Os resultados da reconstrução da odometria são

apresentados na Figura 1-4 onde se faz a comparação dos resultados do sistema de visão estereoscópica com os sensores de odometria. (Saeedi, et al., 2006)

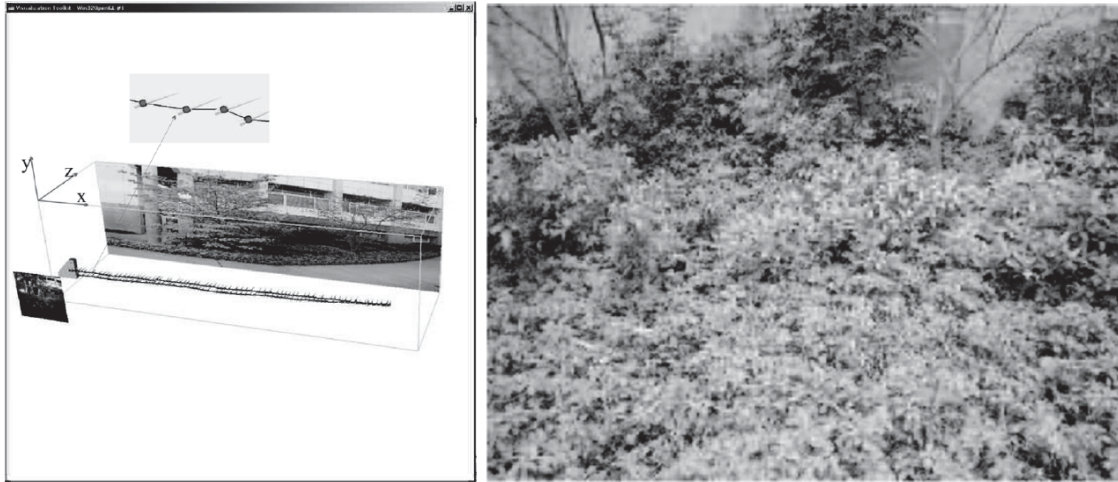


Figura 1-4 Resultados experimentais do trabalho (Saeedi, et al., 2006), ambiente (outdoor), não estruturados.

A navegação SLAM para veículos aéreos, como os apresentados na Figura 1-5, é a habilidade de construir o ambiente a partir das imagens obtidas dos veículos. Nesta área a “University of Freiburg” Alemanha, apresenta um trabalho utilizando duas câmeras.

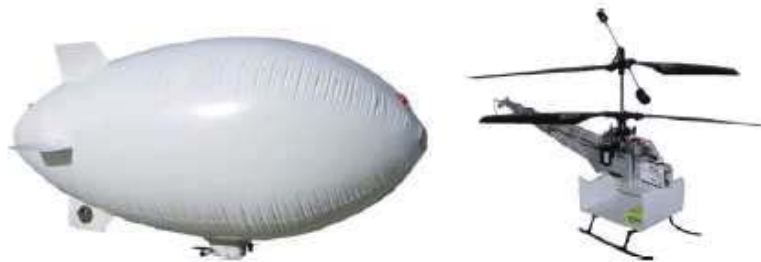


Figura 1-5 Veículo voador não tripulado *Unmanned aerial vehicle* (UAV) (Steder, et al., 2008)

Para a extração de características do ambiente foi utilizado um algoritmo variante do “*progressive sample consensus* (PROSAC)” com maior velocidade e robustez, denominado SURF (*speeded-up robust features*). Foram utilizadas duas câmeras *logitech* como plataforma estereoscópica. Os resultados da navegação e reconstrução são apresentados na Figura 1-6.

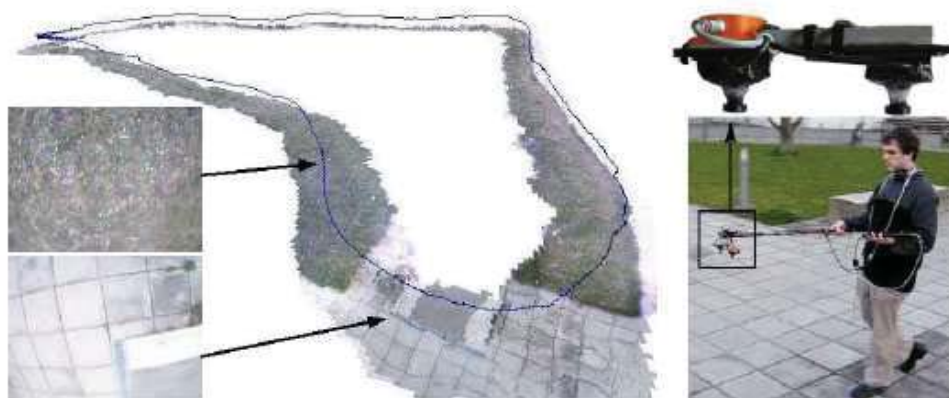


Figura 1-6 Mapa do mundo gerado por visão, câmara estereoscópicas usadas (Steder, et al., 2008)

O sistema foi testado em ambientes internos e externos. Na Figura 1-7 se apresenta a navegação de um veículo autônomo não tripulado, VANT (ou UAV – unmanned aerial vehicle em inglês) em ambientes fechados. (Steder, et al., 2008)



Figura 1-7 Experimento de navegação SLAM em ambientes indoor (Steder, et al., 2008)

Um trabalho desenvolvido na Universidade de Zaragoza, Espanha, utilizou a visão estereoscópica em grande escala com seis graus de liberdade, que consiste em um sistema de localização autônoma e mapeamento (SLAM) em ambientes outdoors e indoors. Esse sistema usa uma câmara estereoscópica com seis graus de liberdade como único sensor (Figura 1-8), apresentando 12 variáveis (posição 3D, orientação com os ângulos de Euler e as velocidades lineares e angulares da câmara). As características de referência foram procuradas nas texturas da imagem. A precisão do algoritmo de reconstrução 3D está em função da distância devido ao uso de diferentes técnicas. Para distâncias curtas foi utilizado

o mapa de disparidade e para distâncias longas utilizou-se a imagem de uma única câmera e alguns pontos, através do algoritmo *inverse depth ID*.



Figura 1-8 Câmera estereoscópica de seis graus de liberdade (Paz, et al., 2008)

O campo de visão em função da distancia é apresentado na Figura 1-9. Os pontos azuis representam pontos reais reconstruídos. A região de incerteza é calculada através da linearização por disparidade avaliada no mundo real e representada na elipse vermelha. A linha verde se refere à incerteza associada à linearização parametrizada ID (Paz, et al., 2008).

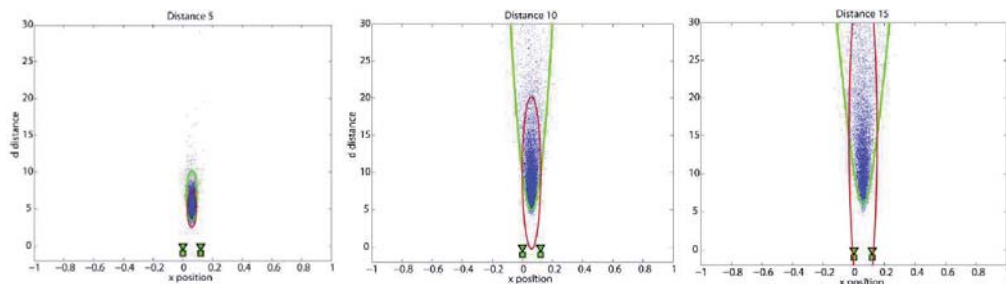


Figura 1-9 Incerteza da visão estérea para diferentes longitudes (Paz, et al., 2008)

O trabalho desenvolvido na Austrália pelo ARC "Centre of Excellence for Autonomous Systems", em colaboração com "Australian Centre for Field Robotics", da Universidade de Sydney, apresenta um algoritmo de localização e mapeamento simultâneo SLAM para a navegação submarina visual de maior escala, no robô *Sirius* apresentado na Figura 1-10. A estimação é baseada nos pontos de vista (*viewpoint augmented navigation (VAM)*), seguido pelo

uso do filtro de Cholesky para filtragem do ruído da imagem. Implementa-se assim uma eficiente estimação de localização de um veículo submarino.(Mahon, et al., 2008)

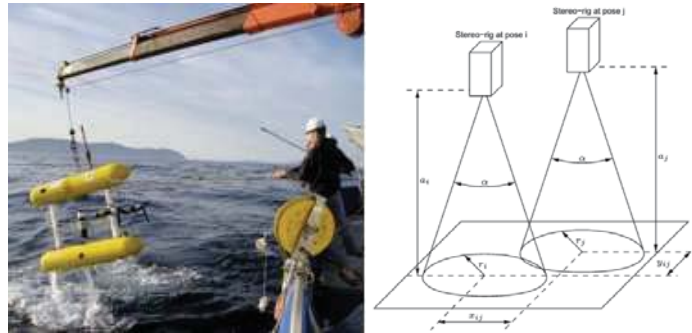


Figura 1-10 Submarino Sirius, sistemas de visão estereoscópica (Mahon, et al., 2008)

Para a estimação da trajetória no submarino Sirius, é implementado um sistema de visão estereoscópica, que utiliza os pontos e a profundidade em instantes consecutivos (i) e (j). Os dois conjuntos de pontos são correlacionados para estimar a trajetória percorrida pelo algoritmo SLAM. No lado esquerdo da Figura 1-11 são comparadas as trajetórias reconstruídas com visão estereoscópica e egocêntrica (“*dead reckoning*”). No lado direito da Figura 1-11 apresentam-se o mosaico das imagens obtidas em dois instantes de tempo e os inter-relacionamentos.

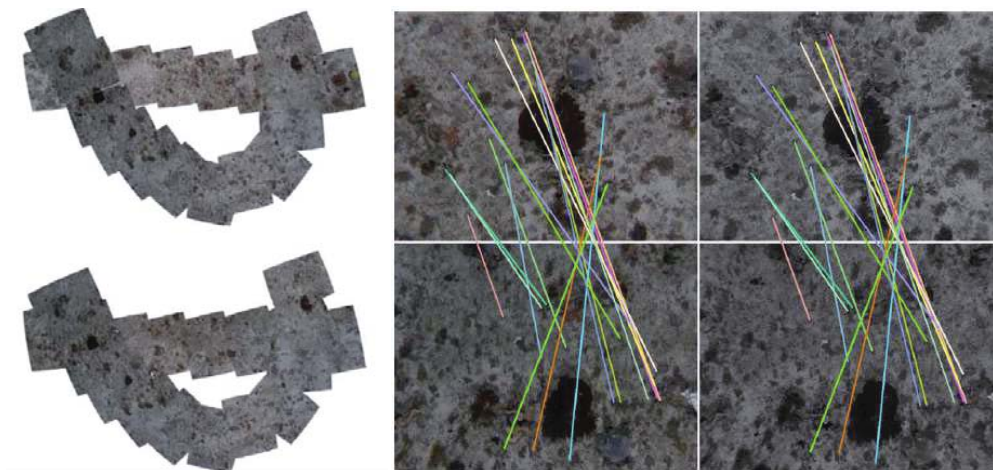


Figura 1-11 Mapa reconstruído por DR e por visão estereoscópica, imagens estéreo em dois tempos (Mahon, et al., 2008)

O tipo de visão nos sistemas vSLAM determina o sistema de coordenadas da reconstrução do ambiente. Os sistemas com visão estereoscópica apresentam a reconstrução 3D em coordenadas euclidianas. Os sistemas de visão monoculares apresentam a reconstrução 3D em coordenadas projetivas.

Um exemplo de visão monocular é o trabalho apresentado por (Silveira, et al., 2007), que desenvolveu um algoritmo de navegação simultânea e reconstrução do ambiente (SLAM), conforme apresentado na Figura 1-12. Esta tarefa geralmente é dividida em três partes. Na primeira, procura-se cuidadosamente as características, utilizado método diretos, empregando a intensidade da imagem com um maior rendimento computacional que outros métodos mais sofisticados da literatura, como o SIFT(Lowe, 2004) ou *Harris detector*(Stephens, 1988). Em seguida correlacionam-se as imagens do instante de tempo atual e a imagem num instante seguinte, associação feita em relação aos pixels, já que com visão monocular a informação sobre profundidade é perdida. Finalmente se estima a trajetória percorrida no espaço projetivo e a reconstrução do ambiente.



Figura 1-12 Reconstrução 3D da trajetória e fotografia do ambiente real (Silveira, et al., 2007)

1.2 Cinemática Do Robô Móvel

As características da movimentação dos robôs móveis são descritas neste capítulo. As possíveis perturbações que interferem na movimentação dos robôs são classificadas. Em seguida é apresentada a modelagem teórica da cinemática de um robô móvel.

1.2.1 Erro de odometria

A odometria localiza o robô, através da acumulação de deslocamentos em um trajeto. A informação de rotação das rodas é sensível a influências do ambiente ou perturbações que geram erros nos registros de deslocamento. Tais erros são classificados em:

Erros sistemáticos	Erros não sistemáticos
Diferenças nos diâmetros das rodas	Trajeto com superfícies desiguais.
Desalinhamento das rodas.	Trajeto sobre obstáculos inesperados na superfície
Limitação na resolução e na frequência de amostra do <i>encoder</i>	Escorregamento das rodas

Tabela 1 Classificação de erros

1.2.2 Modelo Cinemático

Na Figura 1-13 é apresentada a modelagem do robô de duas rodas, com tração diferencial, como o utilizado no presente trabalho. A posição do robô atual é descrita pelo vetor $[x_R, y_R, \theta]$. O ponto preto indica uma posição subsequente do robô ou alvo de movimentação. A distância entre a posição atual e a subsequente é descrita pelos segmentos Δy , Δx , e a orientação de rumo do robô é descrita por α .

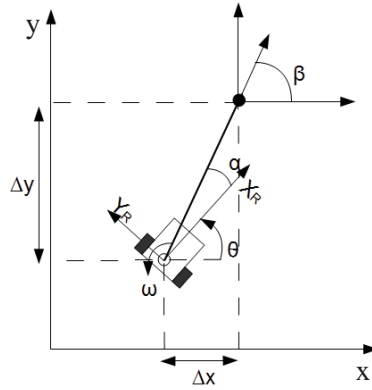


Figura 1-13 Sistema de coordenadas do robô em relação a um objetivo

O modelo cinemático do robô diferencial é descrito na equação (1). A velocidade em direção ao alvo é o vetor $\left[\frac{\Delta x}{t} = \dot{x}, \frac{\Delta y}{t} = \dot{y}, \dot{\theta} \right]$ e a velocidade angular e linear do robô é dada por v e ω :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (1)$$

O modelo em coordenadas polares é representado na Figura 1-14. As equações que descrevem a transformação são.

$$\rho = \sqrt{\Delta x^2 + \Delta y^2}, \quad (2)$$

$$\alpha = -\theta + \text{atan2}(\Delta x, \Delta y) \quad (3)$$

e

$$\beta = -\theta - \alpha. \quad (4)$$

Onde ρ é a distância do centro do robô até o alvo, θ é o ângulo da posição atual do robô, com relação ao eixo X , e β o ângulo formado entre a posição atual do robô e o alvo. A modelagem do sistema em coordenadas cilíndricas é apresentada na Figura 1-14.

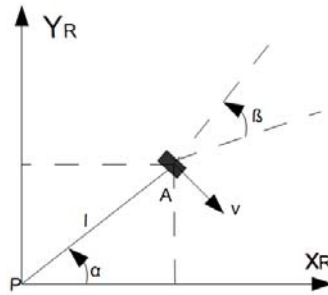


Figura 1-14 Sistema de coordenadas do robô em coordenadas polares

Reescrevendo o modelo cinemático em função das coordenadas polares, equação (5), tem-se:.

$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \frac{-\sin \alpha}{\rho} & 1 \\ \frac{\sin \alpha}{\rho} & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (5)$$

1.3 Objetivos

Com base na revisão dos trabalhos acima, optou-se pela definição do escopo do presente trabalho, representado pelos objetivos descritos abaixo.

Objetivo geral

Localização e navegação de robô autônomo através de visão estereoscópica e odometria.

Objetivos específicos

Identificação das características (*features*) de um ambiente (*indoor*) obtido em duas imagens por um sistema de visão estereoscópica.

Implementação do método de correlação das características encontradas nas imagens, e sua representação tridimensional.

Comparar os resultados com a localização usando a odometria dos encoders.

1.4 Organização do texto

A continuação do presente documento está organizada da seguinte maneira. O Capítulo 2 apresenta os fundamentos teóricos da visão estereoscópica. O Capítulo 3 apresenta a técnica de localização robótica através da odometria. O Capítulo 4 apresenta a plataforma experimental utilizada nas aplicações propostas no presente trabalho. No Capítulo 5 apresenta-se os resultados experimentais de cálculo de odometria visual, discutindo-se os mesmos no Capítulo 6, junto com as conclusões e sugestões para os próximos trabalhos.

2 ESTEREOSCOPIA

Na reconstrução do trajeto feito por um robô móvel são necessárias técnicas e algoritmos da área de processamento de imagens. Nesse capítulo apresentam-se os fundamentos teóricos da visão estereoscópica. Na seção 2.1 são descritos os princípios básicos da calibração de câmera. Na seção 2.2 são identificadas as características correspondentes nas imagens estereoscópicas. Na seção 2.3 é apresentada a reconstrução 3D, através do modelo de visão estereoscópica.

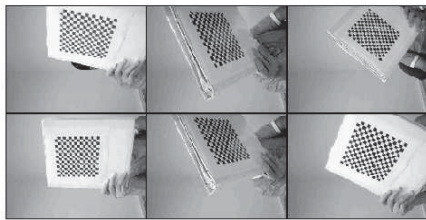
2.1 Calibração da câmera

Em aplicações de fotometria, torna-se necessário a calibração da câmera, técnica onde se estimam os parâmetros intrínsecos e extrínsecos. Os parâmetros extrínsecos representam a movimentação da câmera com relação a um padrão conhecido através de um vetor de translação e uma matriz de rotação, utilizados na conversão de projeções bidimensionais em objetos tridimensionais. Os parâmetros intrínsecos são as características que descrevem cada câmera, representados pela interseção do eixo óptico com o plano da imagem, a distância focal, a distorção da lente e a densidade linear do sensor óptico.

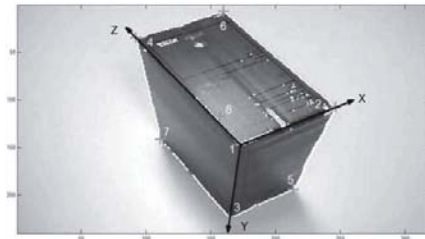
Os algoritmos de calibração são classificados em fotogramétricos e de calibração automática. O primeiro tipo baseia-se na observação de uma cena dinâmica com um objeto de geometria conhecida. O segundo utiliza uma cena estática, com sucessiva movimentação da câmera, capturando imagens em diferentes instantes de tempo. No presente trabalho foram utilizados dois procedimentos fotogramétricos: a calibração de Bouguet (Bouguet, 2003), e um algoritmo de calibração desenvolvido no grupo de pesquisa do DPM (Kurka, et al., 2012).

O procedimento de calibração de Bouguet, disponível como biblioteca pública no ambiente Matlab® ou openCV, utiliza um padrão de calibração quadriculado e plano, com largura e o comprimento conhecidos. Com a finalidade de não gerar posições coplanares a grade quadriculada deve ser deslocada (rotação e translação), como ilustrado na Figura 2-1a. O algoritmo identifica a estrutura geométrica da imagem da grade e encontra os parâmetros intrínsecos e extrínsecos da câmera. Os parâmetros extrínsecos encontrados são: um vetor de translação T e os ângulos de rotação om , com respeito ao sistema de referência da câmera esquerda. A Figura 2-2 mostra uma sequência de imagens do padrão quadriculado usado para a calibração de uma câmera, utilizando o método de Bouguet.

O algoritmo de calibração desenvolvido no âmbito do grupo de pesquisa do DPM/FEM utiliza uma caixa de dimensões conhecidas para estimar os parâmetros intrínsecos e intrínsecos de uma câmera, conforme ilustrado na Figura 2-1b. O algoritmo encontra os vértices da caixa automaticamente, através das propriedades geométricas e algébricas do modelo de projeção da câmera *pinhole*. A Figura 2-3 mostra uma imagem da caixa padrão usada para a calibração de uma câmera, utilizando o método desenvolvido na FEM/DPM Bouguet. Observa-se ainda na mesma figura o sistema de referências e posição dos vértices identificados pelo algoritmo.



(a)



(b)

Figura 2-1 Parâmetros usados na calibração da câmera com Matlab®, (a) apresentação do conjunto de imagens de calibração, (b) Método de calibração de câmera desenvolvido no grupo de pesquisa

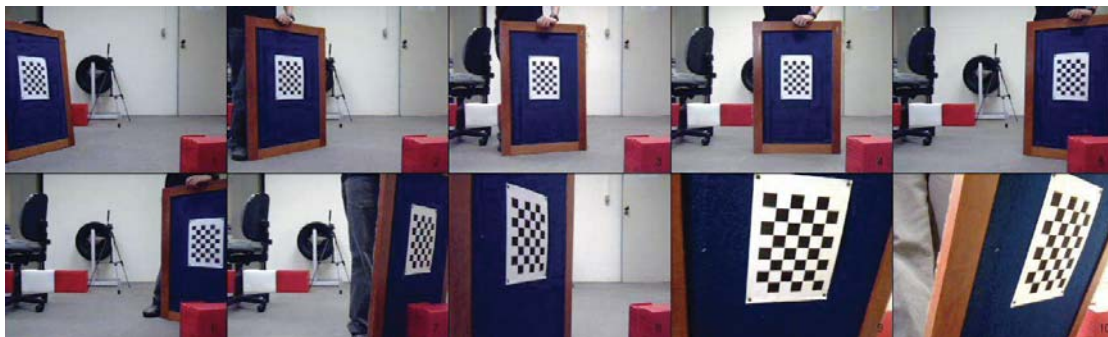


Figura 2-2 Parâmetros usados na calibração da câmera com Matlab®

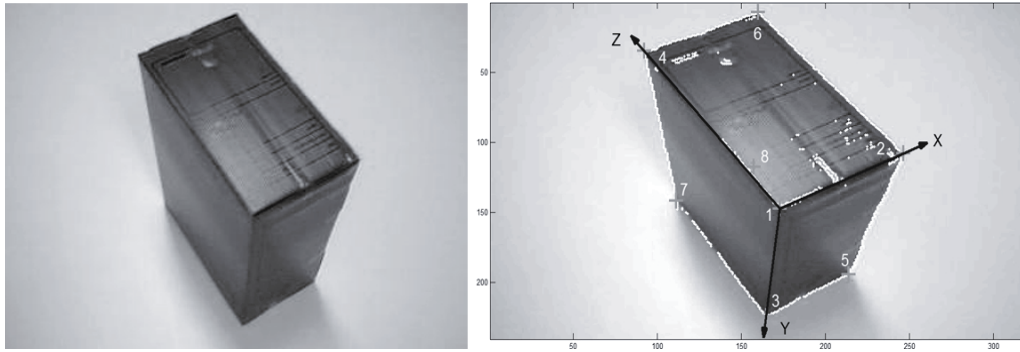


Figura 2-3 Método de calibração de câmera desenvolvido no grupo de pesquisa

2.2 Identificação de características

As características em uma imagem são regiões distintas de pixels (em inglês *keypoints*), referentes a detalhes do ambiente capturado na imagem. A extração de características é um campo onde a eficiência é muito importante, já que o processamento de imagens envolve a execução de um grande número de operações computacionais. Alguns processos requerem um elevado tempo ou capacidade de processamento de um grande número de dados. No presente trabalho, para efeito de localização de pontos especiais de referência de posicionamento do robô, utiliza-se o algoritmo de identificação de características *Scale invariant feature transform* (SIFT), desenvolvido por Lowe (Lowe, 2004), que é caracterizado pela robustez a rotações, mudanças na iluminação ou mudança do ponto de visualização da imagem. O algoritmo SIFT pode encontrar tipicamente, mais de 2000 características, numa imagem de 640x480 pixels. O uso do algoritmo SIFT envolve quatro etapas descritas a seguir:

2.2.1 Detecção dos extremos de escala e espaço

Para a identificar o ponto de interesse é aplicado sobre a imagem um filtro de diferenças gaussianas (DG) com diferentes escalas, com o objetivo de produzir a imagem distorcida L , definida como:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (6)$$

onde

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2 e^{-(x^2+y^2)/2\sigma^2}} \quad (7)$$

sendo σ o fator de distorção, e o símbolo “*” a operação de convolução numérica.

Imagens com diferentes escalas de distorção são obtidas a partir da aplicação da filtragem DG, multiplicado o fator σ por constantes multiplicadoras, definidas em intervalos de oitava, como $k_i = 2^{1/i}$.

Para identificar a localização do ponto estável de interesse, Lowe propôs a diferença de dois espaços de imagens desfocadas em escalas σ e $k\sigma$, definidos como DOG (diferença de gaussianas), ou seja,

$$D_i(x, y, \sigma) = L_i(x, y, \sigma k_i) - L_{i+1}(x, y, \sigma k_{i+1}) \quad (8)$$

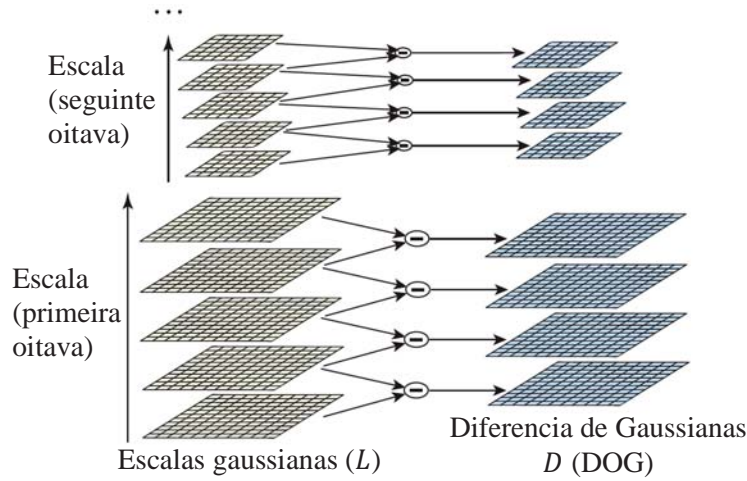


Figura 2-4 Convolução entre as imagens distorcidas L (G) gera a imagem de diferenças gaussianas D (DOG) em cada oitava, (Lowe, 2004).

A diferença de gaussianas DOG pode ser aproximada à equação de calor ou normalização Laplaciana da escala gaussiana G ,

$$\frac{dG}{d\sigma} = \sigma \nabla^2 G \approx \frac{G_i - G_{i+1}}{k_i - k_{i+1}} \cdot \frac{1}{\sigma}. \quad (9)$$

Isolando a diferença gaussiana tem-se,

$$\sigma^2 \nabla^2 (k_i - k_{i+1}) \approx G_i - G_{i+1}, \quad (10)$$

onde o fator $(k_i - k_{i+1})$ e σ são constantes sobre toda a escala e não influenciam a localização do ponto. O critério utilizado para a localização do ponto é a magnitude do gradiente ∇ .

2.2.2 Localização do keypoint

Para a busca do ponto de interesse (*keypoint* no ambiente Sift) é analisado o gradiente ∇ das imagens D_i . A busca é executada em cada pixel da imagem. Inicia com a comparação do comprimento do gradiente em uma vizinhança Moore 3D, com raio 1. Uma vizinhança de Moore com raio 1 é definida como: Imagem D_{i+1} (8_{pixels})+ D_{i+1} (9_{pixels})+ D_i (9_{pixels})+imagem D_{i+2} , conforme iilustrado na Figura 2-5. Os pontos que são seleccionados como candidatos a *keypoints* possuem um gradiente maximo ou minimo entre os 26 pixel vizinhos.

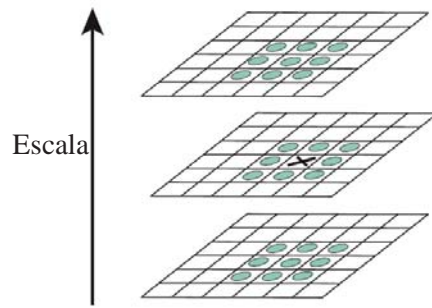


Figura 2-5 Vizinhança 3D de Moore do *keypoint* com radio 1 (Lowe, 2004).

Depois de identificar são descartados os *keypoint* candidatos que tenham baixo contraste ou estejam alocado sobre as bordas (edges).

2.2.3 Orientação do keypoint

A orientação de cada *keypoint* é baseada nos gradientes locais de orientação e magnitude em uma imagem local distorcida L_i , onde i corresponde a escala do *keypoint*. O procedimento inicia estimando os gradientes de orientação $\theta(x, y)$ e magnitude $m(x, y)$ através de,

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2} \quad (11)$$

e

$$\theta(x, y) = \tan^{-1} \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}. \quad (12)$$

Um histograma de orientação é construído pelos gradientes de orientação no *keypoint* e na sua vizinhança, de acordo com (Lowe, 2004). A orientação é representada em um vetor de 36 posições correspondente aos 360 graus de uma circunferência, divididos em intervalos de dez graus. Cada amostra inserida no histograma é ponderada pelo gradiente de magnitude m e por uma janela gaussiana circular com $\sigma_j = 1,5\sigma k_i$. Finalmente a direção é definida através de dois *keypoints*. Um selecionado no pico do histograma de orientação e outro correspondente a 80% do valor do pico.

A identificação do local e a orientação de um *keypoint* com o algoritmo SIFT, apresenta uma repetibilidade de 80-95%, em configurações de baixo ruído ($< 2\%$) e ângulos de inclinação menores do que 20 graus. No presente trabalho foi utilizada a implementação do SIFT do trabalho de Vedaldi (A. Vedaldi, 2010). Resultados típicos de obtenção dos *keypoints* pelo algoritmo SIFT e seus descritores, são apresentados na Figura 2-6.



Figura 2-6 Resultados experimentais do algoritmo SIFT

2.3 Visão estereoscópica

2.3.1 Geometria epipolar

O sistema de visão estereoscópica proposto no presente trabalho é composto por duas câmeras. A modelagem teórica é representada por dois sistemas da câmera *pinhole* conforme ilustrado na Figura 2-7. Os centros das câmeras O_l e O_r são ligados por uma linha chamada linha base. A interseção da linha base com os planos de projeção geram os epipolos e_l e e_r , ou seja, e_r é a projeção do centro O_l na imagem da direita. A visualização de um ponto P no espaço gera projeções nas duas imagens p_r e p_l . As linhas que ligam os epipolos com as projeções são chamadas linhas epipolares l_l e l_r .

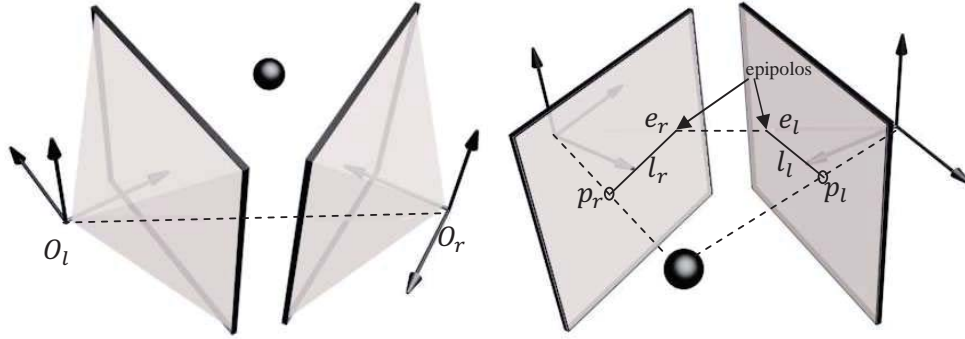


Figura 2-7 Geometria epipolar de duas câmeras, vistas frontal e traseiras.

A relação entre o ponto de projeção e o ponto no espaço é a matriz \mathbf{M} de parâmetros intrínsecos e extrínsecos.

$$p_l = \mathbf{M}_l P \quad \text{e} \quad p_r = \mathbf{M}_r P \quad (13)$$

A reconstrução de um ponto no espaço da informação projetada é descrita pela seguinte equação.

$$P = \mathbf{M}^+ p \quad (14)$$

Onde \mathbf{M}^+ é a pseudo-inversa de \mathbf{M} . Os centros de cada câmera são projetados na imagem da câmera oposta e a matriz da câmera \mathbf{M}_l cumpre a equação $\mathbf{M}_l O_r = e_l$.

A linha (O_r, P) projetada na câmara esquerda na linha epipolar l_l , que passa pelos pontos e_l e p_l pode-se escrever no produto vetorial \times ou produto vetorial matricial $S()$.

$$l_l = e_l \times (\mathbf{M}_l \mathbf{M}_r^+ p_r) = S(e_l) \mathbf{M}_l \mathbf{M}^+ p_r \quad (15)$$

A matriz fundamental representa o mapeamento do ponto p_r na linha epipolar correspondente e definida por,

$$\mathbf{F} = S(e_l) \mathbf{M}_l \mathbf{M}^+ \quad (16)$$

2.3.2 Reconstrução 3D

O modelo para a reconstrução 3D com rotação entre as câmeras é apresentado na Figura 2-8. Dois sistemas de coordenadas representam os centros das câmeras. A rotação e translação são definidas pelo vetor T e a matriz R .

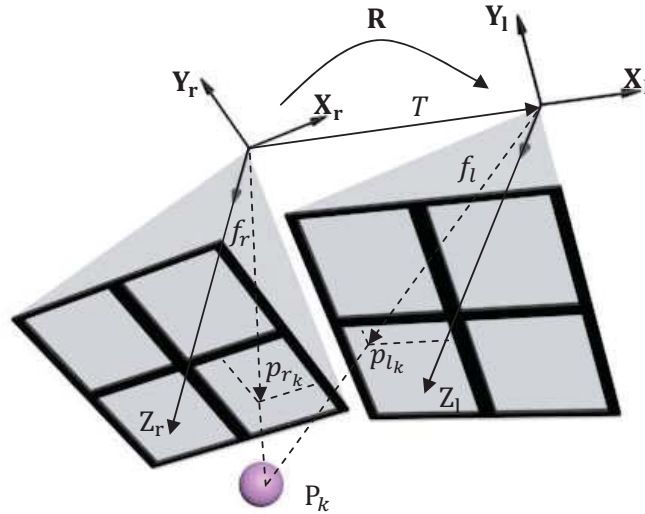


Figura 2-8 Projeção de um ponto em um sistema estereoscópico

As coordenadas espaciais de profundidade z_{lk} e z_{rk} para um conjunto de pontos no espaço são obtidas dos vetores p_{lk} e p_{rk} em coordenadas de câmera.

$$\mathbf{Z}_l = (\mathbf{E}[\mathbf{p}_{lk} \quad -\mathbf{R}\mathbf{p}_{rk}])^{-1}\mathbf{E}T \quad (17)$$

Onde

$$\mathbf{Z}_k \equiv \begin{bmatrix} z_{lk} \\ z_{rk} \end{bmatrix} \quad (18)$$

e

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \quad (19)$$

Os vetores com a posição espacial das características ts com referência as câmeras direita e esquerdas são expressas como:

$$\mathbf{P}_{lk} = \frac{z_{lk}}{f_l} \mathbf{p}_{lk} \quad \mathbf{P}_{rk} = \frac{z_{rk}}{f_r} \mathbf{p}_{rk}, \quad (20)$$

Onde f_l e f_r são a distância focal da câmera esquerda e direita, respectivamente.

2.3.3 Disparidade

Quando a rotação entre as duas câmeras é zero, a geometria estereoscópica é conhecida como disparidade. Na Figura 2-9 apresenta-se o modelo pinhole para duas câmeras paralelas. A linha base é denotada com bl , os centros de projeção são O_l e O_r , a distância do centro até o plano de projeção é definido pelo parâmetro intrínseco "distância focal" f . O ponto P é representado pelas projeções p_r e p_l e a distância dos pontos de projeção até os centros das imagens c_r e c_l , são x_l e x_r .

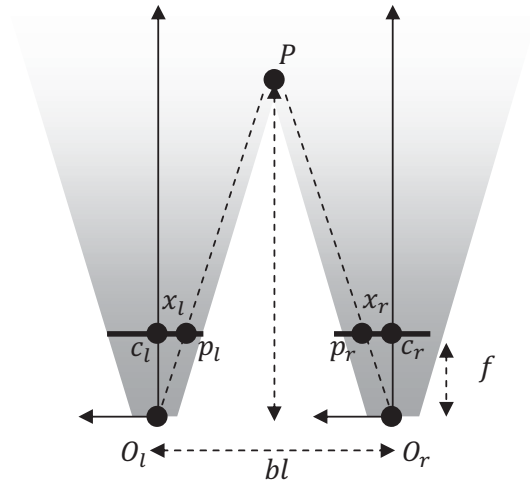


Figura 2-9 Sistema de visão estereoscópica com câmeras paralelas.

A disparidade d tem como objetivo, estimar a profundidade do ponto P no espaço, definida como a diferença entre as projeções do ponto P e centro da imagem,

$$d = x_l - x_r < 0 \quad (21)$$

O sistema global de coordenadas do sistema estereoscópico neste trabalho é definido no sistema da câmera esquerda. Utilizando o modelo da câmera *pinhole* da seção 2.3.1, é definido o ponto $P = [x, y, z]$ em função das projeções e os parâmetros da câmera.

$$\frac{p_r}{f} = \frac{\frac{bl}{2} + x}{z}, \quad \frac{p_l}{f} = \frac{x}{z} \quad (22)$$

Isolando x optem-se a relação,

$$\frac{p_l}{f} z = \frac{p_r}{f} z - \frac{bl}{2} \quad (23)$$

Onde a profundidade é,

$$z = \frac{blf}{2(p_r - p_l)} = \frac{blf}{2d}. \quad (24)$$

2.3.4 Correspondência

O problema de correspondência consiste em determinar pontos correspondentes entre duas imagens. A maior parte de pontos no espaço são visíveis pelas duas câmeras e algumas regiões das duas imagens são similares. A complexidade do ambiente, as mudanças de iluminação fazem com que a correspondência de imagens se converta em um problema de busca.

No presente trabalho a busca de pontos correspondentes é executada em quatro imagens. Dois pares de imagens estereoscópicas são armazenados em posições subsequentes. O procedimento é apresentado na Figura 2-10. A primeira busca procura pontos correspondentes entre as imagens a esquerda (L), nas posições anterior $t(i - 1)$ e atual $t(i)$, estes pontos são procurados nas imagens a direita (R) na busca 2. A última busca (3), compara os pontos resultantes da busca 2, nas posições anterior e atual.

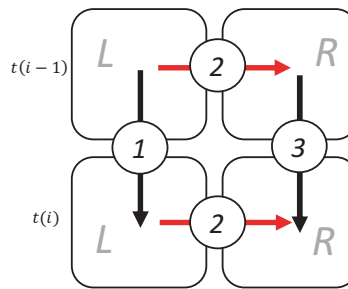


Figura 2-10 Correspondência nas imagens

3 LOCALIZAÇÃO PELA ODOMETRIA VISUAL

Neste capítulo é apresentada a técnica de odometria visual. Descreve-se, inicialmente, o sistema utilizado composto pela câmera estereoscópica e o robô. Depois, apresenta-se a modelagem matemática para a recuperação do trajeto através de dois conjuntos de vetores. Finaliza-se o processo com a apresentação da trajetória, por meio de sistemas de coordenadas consecutivos.

Uma das técnicas de localização em robótica é a odometria. Esta consiste em estimar a posição do robô no ambiente através da acumulação de dados de deslocamento. A metodologia comumente utilizada estima o deslocamento do robô, através da rotação e o raio nas rodas, utilizando a informação dos encoders ao longo do tempo. Neste trabalho é utilizada uma metodologia visual através da informação de imagens captadas em duas câmeras. O sistema de odometria visual utilizado é apresentado na Figura 3-1.

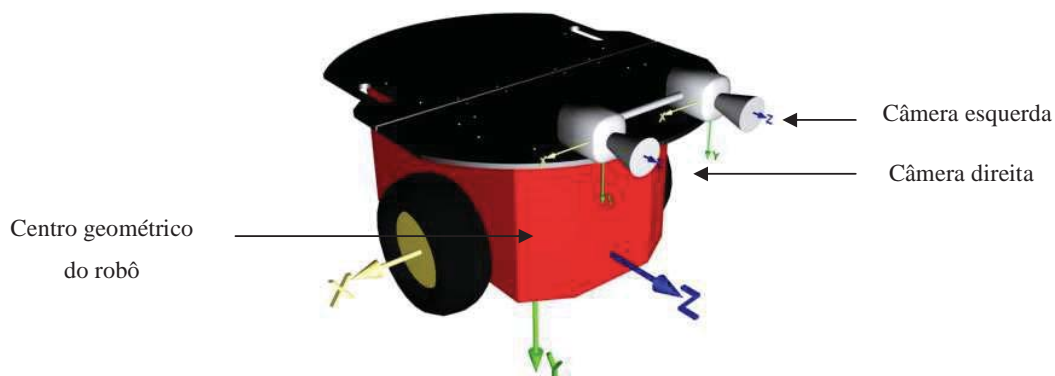


Figura 3-1 Sistema de coordenadas do robô e das câmeras

A técnica de visão estereoscópica (seção 2.3) é baseada na medição e observação de pontos no espaço. Com as medições em dois instantes de tempo, são construídos dois conjuntos de pontos correspondentes. Na Figura 3-2 são apresentados dois conjunto de vetores, reconstruídos no espaço euclidiano.

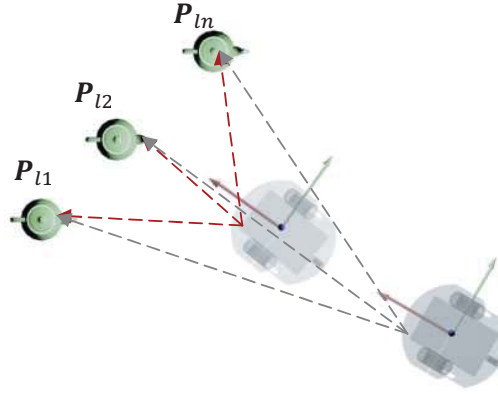


Figura 3-2 Vetores da câmera direita em dois instantes de tempo.

A estimação da movimentação no espaço requer um número mínimo de sete pontos correspondentes nos dois instantes de tempo. O deslocamento é representado por um vetor de translação \mathbf{T}_{mi} e uma matriz de rotação \mathbf{R}_{mi} , apresentadas nas Figura 3-3.

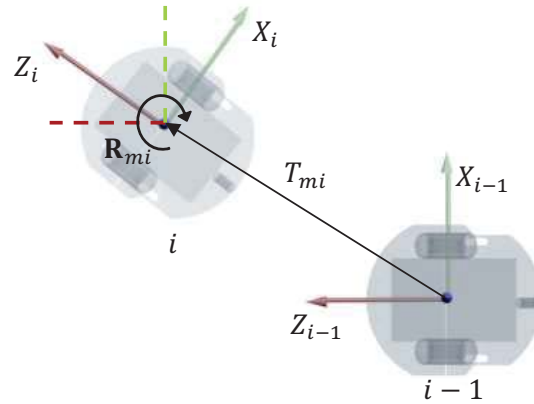


Figura 3-3 Deslocamento do robô

Os conjuntos de pontos, reconstruídos no espaço euclidiano, são armazenados nas matrizes \mathbf{P}_i e \mathbf{P}_{i-1} de tamanho $4 \times n$ descritas como,

$$\mathbf{P}_{i-1} = \begin{bmatrix} \mathbf{P}_{l1}(i-1) & \mathbf{P}_{l2}(i-1) & \dots & \mathbf{P}_{ln}(i-1) \\ 1 & 1 & \dots & 1 \end{bmatrix}$$

$$\mathbf{P}_i = \begin{bmatrix} \mathbf{P}_{l1}(i) & \mathbf{P}_{l2}(i) & \dots & \mathbf{P}_{ln}(i) \\ 1 & 1 & \dots & 1 \end{bmatrix}, \quad (25)$$

onde $P_{lk}(i)$ representa a posição espacial do keypoint “k” com relação a câmera esquerda, calculada no instante i , achando n keypoints correlatos nos instantes $i - 1$ e i . A matriz de transformação T_{RTi} que contém a movimentação (matriz de rotação e vetor de translação), é estimada com as matrizes P_{i-1} e P_i como,

$$T_{RTi} = P_{i-1} P_i^+ , \quad (26)$$

onde “+” representa a operação pseudo inversa, e

$$T_{RTi} = \begin{bmatrix} R_{mi} & T_{ni} \\ 0 & 1 \end{bmatrix}. \quad (27)$$

A posição atual do robô, é definida como vetor no referencial *home*, descrita como

$$X_i = \prod_0^i (R_{m(i-1)} X_{i-1}) + T_{mi} , \quad (28)$$

com

$$X_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \quad (29)$$

A estimação da odometria é feita com o calculo sucessivo das rotações e translações do robô como apresentado na Figura 3-4.

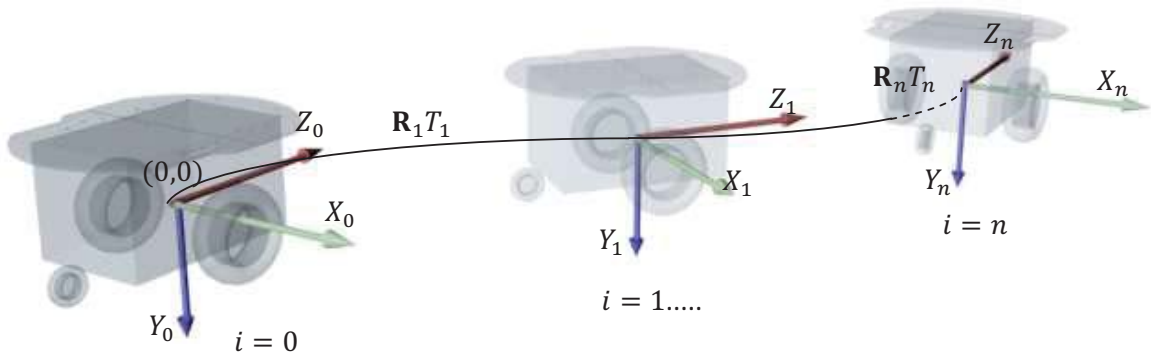


Figura 3-4 Odometria de sistemas de coordenadas consecutivas

4 DISPOSITIVOS USADOS NA IMPLEMENTAÇÃO DOS MÉTODOS DE LOCALIZAÇÃO

Os equipamentos utilizados para a navegação e localização do robô são apresentados neste capítulo. O robô móvel Pioneer 3D-X é descrito em detalhes e as câmeras estereoscópicas utilizadas são apresentadas.

4.1 Robô Móvel Pioneer 3D-X

O robô Pioneer 3D-X Figura 4-1, desenvolvido pela empresa Mobile Robots, faz parte de uma família desenvolvida para pesquisa. O modelo utilizado é do tipo diferencial com tração em duas rodas. Contém uma CPU embarcada, diversos sensores e software de controle e comunicação. As principais características de robô são:

- Duas rodas com tração diferencial.

- 16 sonares com o alcance de 0,15m a 5m.

- Um microcontrolador 16-bit Hitachi H8S.

- Um computador embarcado Pentium 3 (Pentium M).

Os acessórios que o compõem são:

- Um laser de 180 graus,

- Um gripper

- Parachoques (*Bumpers*), com sensores de final de curso.

- Câmera estereoscópica (anexa para o desenvolvimento do projeto)



Figura 4-1 Robô Móvil Pioneer 3D-X

A arquitetura do robô é composta por dois níveis de sistemas, que são apresentadas na Figura 4-2. Os subsistemas Aria, HTTPhthru e a API do Matlab representam um conjunto de algoritmos de alto nível. No conjunto de baixo nível está o subsistema encarregado do controle do hardware ARCOS.

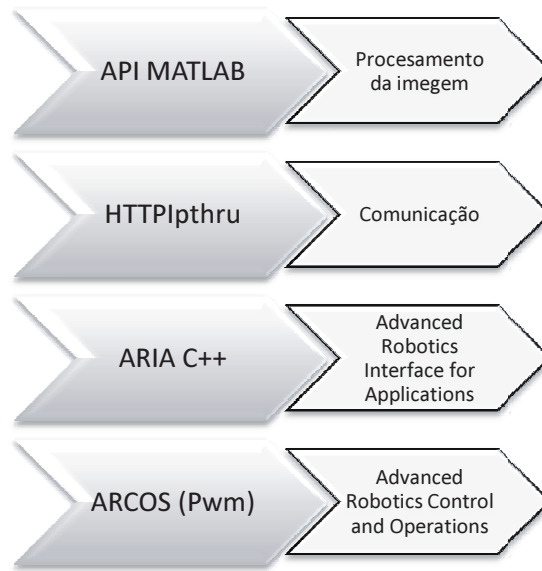


Figura 4-2 Arquitetura do Robô Móvil Pioneer 3D-X

4.2 Câmera estereoscópica

Duas câmeras estereoscópicas são utilizadas neste trabalho. A primeira câmera foi modelada virtualmente para testar os algoritmos de calibração e reconstrução 3D, utilizando a ferramenta de modelagem tridimensional 3DsMax. A outra é um protótipo real, construído com câmeras tipo *web-cam*.

4.2.1 Simulação da câmera estereoscópica com 3DsMax

A simulação do sistema de visão utiliza três modelagens tridimensionais. A primeira é a modelagem da câmera estereoscópica, representada por duas câmeras virtuais. A segunda

modelagem representa o robô Pioneer com a câmera estereoscópica. Neste modelo são representadas a geometria real e as restrições de locomoção. Finalmente, numa terceira modelagem, o ambiente de navegação *indoors* é simulado.

O modelo da câmera estereoscópica é apresentado na Figura 4-3. O sistema de coordenadas da câmera esquerda é o sistema global.

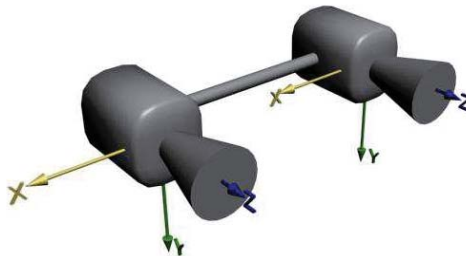


Figura 4-3 Modelo da câmera estereoscópica

O robô Pioneer virtual é modelado como indicado na Figura 4-4. O centro geométrico do robô é o referencial para obter o posicionamento no espaço. Esta informação é o referencial de simulação da variável de posição dependente dos encoders no robô real.

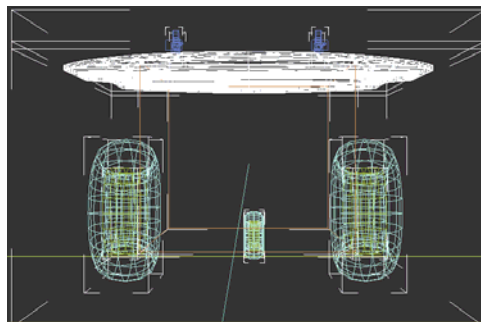


Figura 4-4 Modelo 3D do robô Pioneer

Dois ambientes virtuais são construídos, um para a calibração e outro para a navegação. O uso de texturas e luzes é fundamental para que o ambiente tenha a informação visual, e dessa forma, podem ser identificados os *keypoints* através dos algoritmos de processamento de imagem. O ambiente de calibração virtual é apresentado na Figura 4-5.



Figura 4-5 Ambiente 3D de calibração

A caixa de calibração é visualizada em doze perspectivas diferentes. Desta forma são originadas as imagens utilizadas para calcular os parâmetros intrínsecos e extrínsecos da câmera estereoscópica. As imagens obtidas na câmera esquerda são apresentadas abaixo na Figura 4-6.

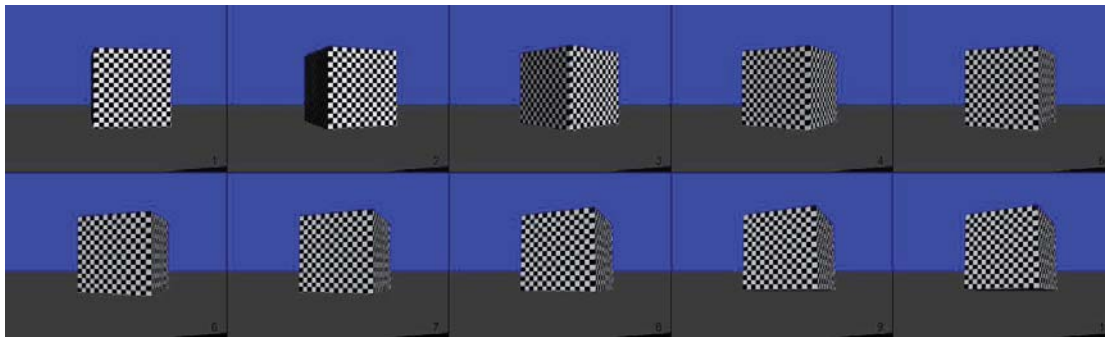


Figura 4-6 Imagens da caixa de calibração

Após a modelagem do robô e do ambiente de calibração, apresenta-se a modelagem do ambiente de navegação. O ambiente virtual é representado por móveis que compõe um laboratório. Os objetos representados paredes, pisos, armários e cadeiras têm texturas que acrescentam a informação visual e facilitam a identificação das características (*keypoints*). As texturas e o modelo de corredor utilizado são apresentados abaixo na Figura 4-7. Maiores detalhes do ambiente virtual de deslocamento do robô são apresentados no capítulo 6.



Figura 4-7 Texturas do modelo, modelo do corredor do laboratório

4.2.2 Construção da câmera estereoscópica

O sistema de visão estereoscópica é composto de duas *webcams* “Logitech C160” (Figura 4-8) e uma base metálica que liga este sistema ao robô.



Figura 4-8 Câmera Webcam Logitech C160

A base de fixação da câmera esquerda foi projetada para ser montada no robô Pioneer. Finalmente, a câmera estereoscópica montada é apresentada na Figura 4-9 abaixo.



Figura 4-9 Câmera estereoscópica

A área de trabalho da câmera estereoscópica é definida pela interseção do ângulo de visão (FOV 45°) de cada câmera. As duas câmeras estão afastadas de 19 cm, entre centros. A área de trabalho, tem um formato triangular, com uma distancia mínima de 223mm, e um ângulo de 46° , representado pela área hachurada na Figura 4-10.

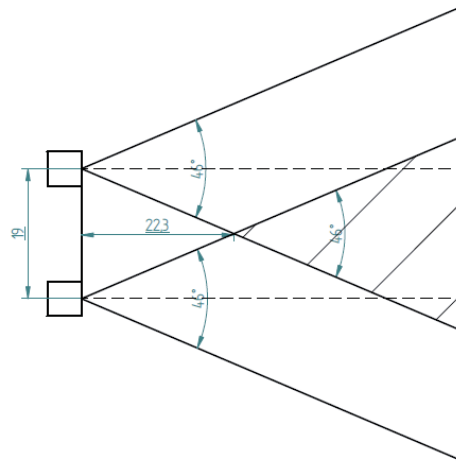


Figura 4-10 Ângulo de vista da câmera estereoscópica

5 RESULTADOS

Este capítulo apresenta a implementação dos elementos descritos na modelagem teórica, através dos experimentos feitos em ambiente virtual e real, utilizando-se a ferramenta de modelagem tridimensional 3D–Max e o robô móvel Pioneer.

A seção 5.1 apresenta a acurácia e o erro do sensor óptico proposto no presente trabalho (câmera estereoscópica). A seção 5.2 aborda o problema da reconstrução de um ambiente a partir de informações geométricas capturadas por sensores embarcados em um robô. Apresenta-se o procedimento proposto de reconstrução 3D utilizando um sistema de câmeras estereoscópicas. Finalizando, a seção 5.3 apresenta as provas experimentais de navegação e localização de uma trajetória, utilizando a câmera estereoscópica e o robô Pioneer 3D-X.

5.1 Calibração estereoscópica utilizando o algoritmo de Bouguet

Com a calibração de cada câmera individualmente (seção 2.1), é possível estimar os parâmetros extrínsecos da câmera estereoscópica. O sistema de calibração estéreo considera a câmera da esquerda como sistema de referências global. O passo seguinte é a reconstrução 3D do ambiente das duas câmeras e a estimação dos parâmetros extrínsecos. Para isto usou-se o módulo de “*Extrinsic Parameters*” do toolbox de calibração de Matlab. Na Figura 5-1 pode-se ver a reconstrução das posições dos padrões quadriculados apresentadas para diferentes ângulos com a translação e rotação de cada câmera.

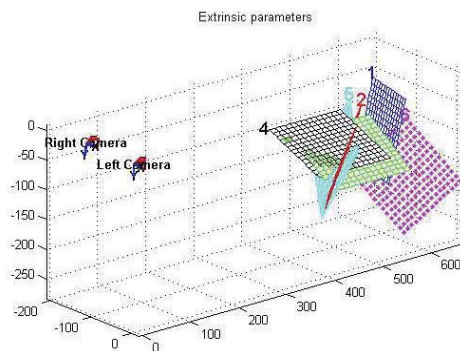


Figura 5-1 Reconstrução 3D das grilas e a locação das câmeras no espaço.

Os parâmetros encontrados apresentam um vetor de translação T e os ângulos de rotação om , estes se encontram em relação do sistema de referência da câmera esquerda.

Os parâmetros extrínsecos (precisão da câmera direita com relação à câmera esquerda) foram calculados como:

Vetor de rotação: $om = [-0.05773 \quad -0.17908 \quad -0.00840]$

Vetor de translação: $T = [841.12860 \quad -0.50898 \quad 108.07729]$

Para estimar a precisão do sistema de câmeras estereoscópicas é medido o erro do algoritmo de reconstrução 3D, do qual foram realizados diversos testes. Inicialmente utilizou-se um alvo (bola laranja apresentada na Figura 5-2), medindo-se a distância desta até a câmera, com uma fita métrica e com o sistema de visão estereoscópica. As medidas são comparadas indicando a margem de erro do sistema de visão.

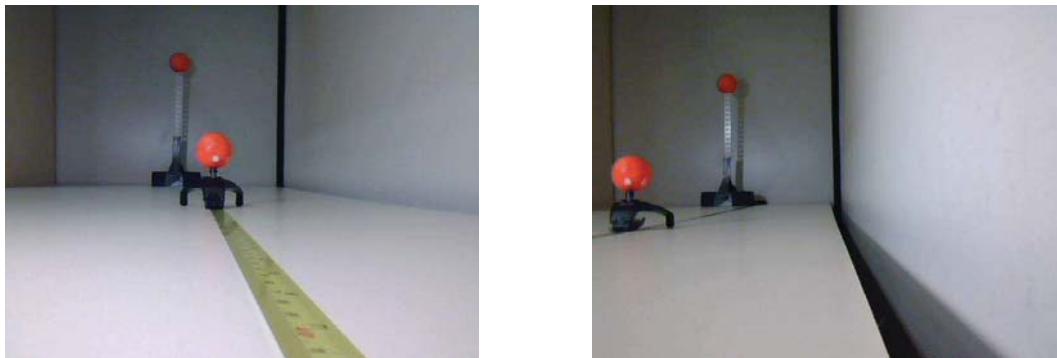


Figura 5-2 Imagens estereoscópicas para testar a calibração

A distribuição normal do erro encontrado entre os dois sistemas de medição, para um universo de 100 medidas, com um erro médio de 3,7% é apresentado abaixo na Figura 5-3.

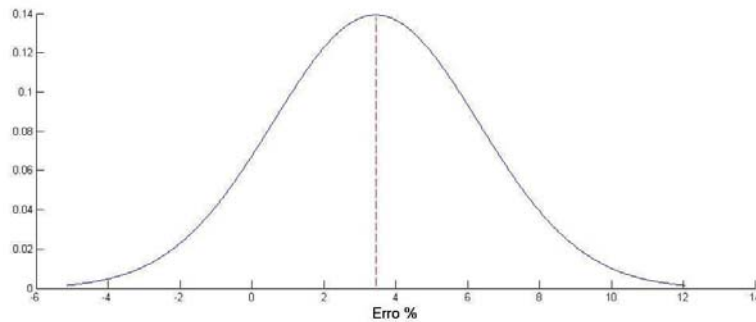


Figura 5-3 Distribuição normal da câmera estereoscópica

5.2 Reconstrução do ambiente

A presente seção trata do problema de reconstrução de um ambiente a partir de informações geométricas captadas por sensores embarcados no robô. Apresenta-se o procedimento proposto de reconstrução 3D utilizando um sistema de câmeras estereoscópicas.

O segundo problema da visão estereoscópica é a reconstrução do ambiente em três dimensões (Emanuele Trucco, 1998). Depois de se obter uma correlação de pontos entre as imagens estereoscópicas, e as imagens em cada deslocamento, é feita a reconstrução da posição espacial de tais pontos. Inicia-se com a translação de cada ponto, ao sistema de coordenadas global. Utiliza-se o procedimento de reconstrução da odometria descrito no capítulo 0, ou seja, foi multiplicado cada ponto pelas matrizes de rotação e os vetores de translação dos sistemas anteriores. Em uma experiência de deslocamento linear com o robô, reconstruíram-se os pontos utilizados em cada intervalo. Os pontos vermelhos, na figura 6-7 abaixo, representam o conjunto de pontos no espaço, identificados a partir de uma posição do robô em um determinado instante de tempo. Os pontos verdes representam o mesmo conjunto de pontos especiais, identificados pelo robô em uma posição e instante de tempo subsequente. Também na Figura 5-4 é apresentado o trajeto feito pelo robô, reconstruído por estimativas de deslocamento anteriores.

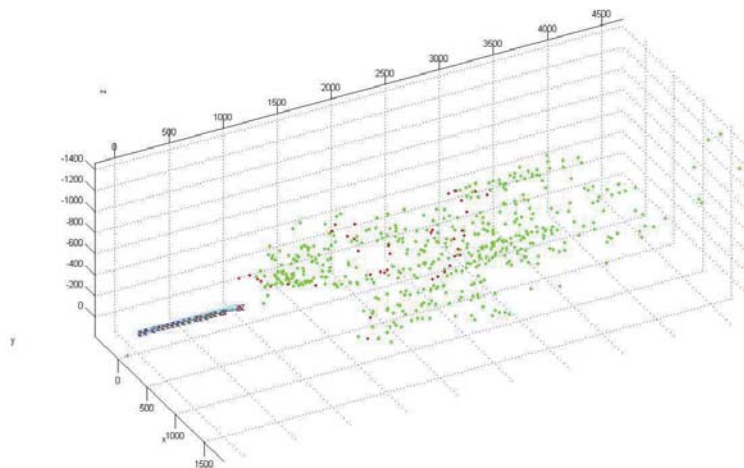


Figura 5-4 Reconstrução 3D do ambiente

5.3 Odometria visual

O algoritmo de odometria visual foi testado no laboratório, em um ambiente de interiores (*indoor*), com os deslocamentos do robô em uma superfície plana (eixos z e x). Foram feitos dois tipos de trajetos, o primeiro, uma linha reta, e o segundo na forma de uma senoide. Cada movimento é dividido em pequenos trechos retilíneos onde o deslocamento é controlado pelo algoritmo descrito na seção 4.1.

A representação da trajetória percorrida é composta por sistemas de coordenadas, que variam a origem com relação ao instante anterior, reconstruindo a odometria. A Figura 5-5 abaixo representa um deslocamento no eixo Z .

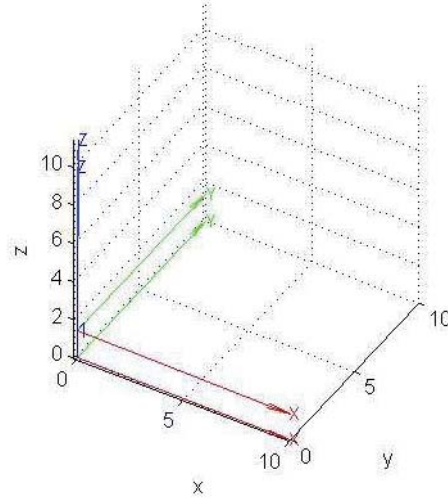


Figura 5-5 Representação dos sistemas de coordenadas

Os experimentos de recuperação da odometria podem ser divididos em três etapas. Na primeira, a variável de posição inicial é considerada como zero e nomeada *home*. Em uma segunda etapa, é gerada uma lista de sub-alvos através da divisão do trajeto em seções iguais. Em uma terceira etapa, o robô avança ao seguinte sub-alvo, continuando até completar o deslocamento global.

5.3.1 Filtro de correlações

As provas apresentaram baixa sensibilidade do algoritmo de estimação da odometria, devido à correlação incorreta de pontos nas diferentes imagens, o que exigiu a implementação de uma etapa para descartar correlações incorretas.

O filtro implementado é do tipo on-off, o qual descarta os valores que estão fora da área definida pelos pontos de corte. Na Figura 5-6 é apresentada a maneira de exemplo a filtragem de um sinal. A linha contínua vermelha representa o sinal. Os círculos azuis representam a resposta do filtro, alocando em zero os pontos descartados e no valor médio do sinal os não descartados. A área de aceitação, definida em função da intensidade (0-100%) do filtro é descrita pelo intervalo $[V_{medio}(1 - intensidade), V_{medio}(1 + intensidade)]$.

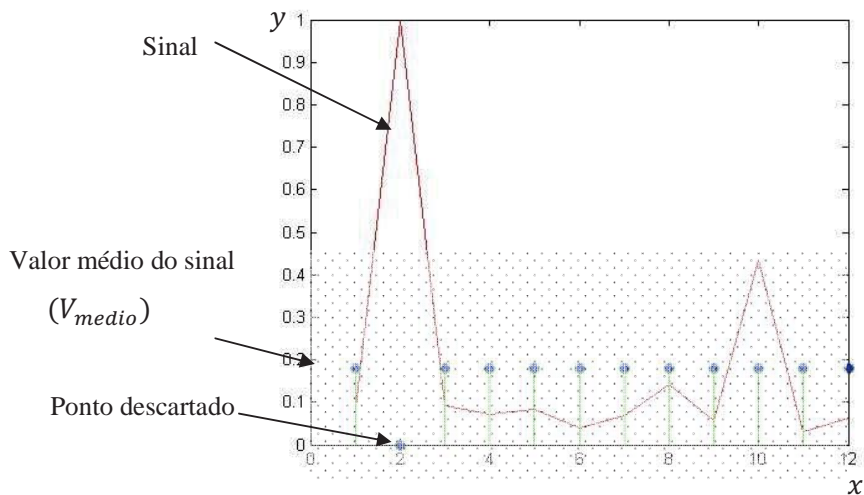


Figura 5-6 Representação dos pontos descartados pelo filtro

5.3.2 Trajetória Linear

O algoritmo de navegação e localização em uma trajetória linear é testado em um ambiente virtual e real. O ambiente de modelagem virtual utiliza a plataforma Autodesk 3D-MAX, simulando superfícies planas como paredes, tetos e alguns objetos comumente encontrados em um escritório. Na Figura 5-7 são mostradas as imagens capturadas pelas câmeras estereoscópicas na navegação virtual.



Figura 5-7 Imagens da simulação do ambiente virtual

As correlações das imagens no ambiente simulado são apresentadas abaixo, onde os vetores ligam os pontos que foram encontrados nos dois instantes de tempo. As correlações erradas ligam

pontos que não correspondem ao deslocamento, estas fazem com que a estimação da rotação e translação local apresentem erros.

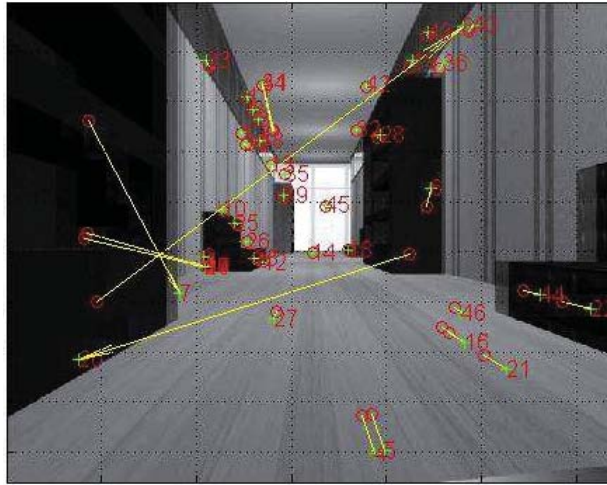


Figura 5-8 Correlações da imagem da direita no ambiente virtual

A trajetória reconstruída, da simulação no ambiente de modelagem 3D-max, é apresentada abaixo. O algoritmo apresenta um erro incremental, afastando a trajetória em 10% da posição final.

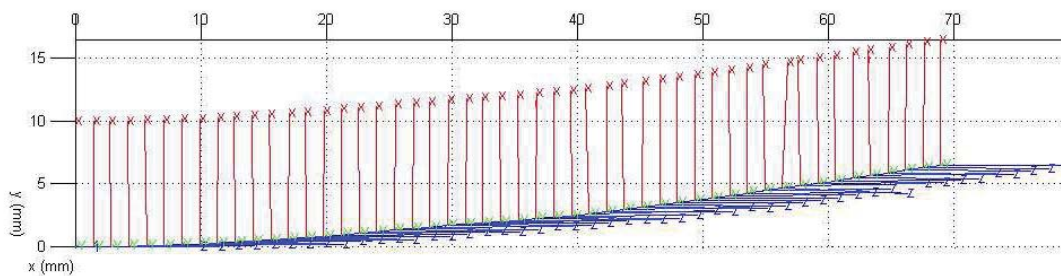


Figura 5-9 Trajetória reconstruída da simulação no ambiente virtual

A trajetória proposta para o experimento real é uma trajetória reta de um metro de comprimento no eixo z, é dividida em 29 deslocamentos. Tal discretização é feita com a intenção de deter o robô para obter a informação dos sensores e processar as imagens e finalizar com a estimação da odometria visual. O ambiente e o deslocamento do robô no laboratório são apresentados nas imagens Abaixo.



Figura 5-10 Deslocamentos no eixo Z, ambiente real

Cada câmera gera uma imagem de 1.3 Mgps e o algoritmo Sift reconhece os pontos mais importantes e correspondentes nas duas câmeras. Algumas das imagens da câmera esquerda, utilizadas para a reconstrução da trajetória com o algoritmo de odometria visual são apresentadas na Figura 5-11 abaixo.

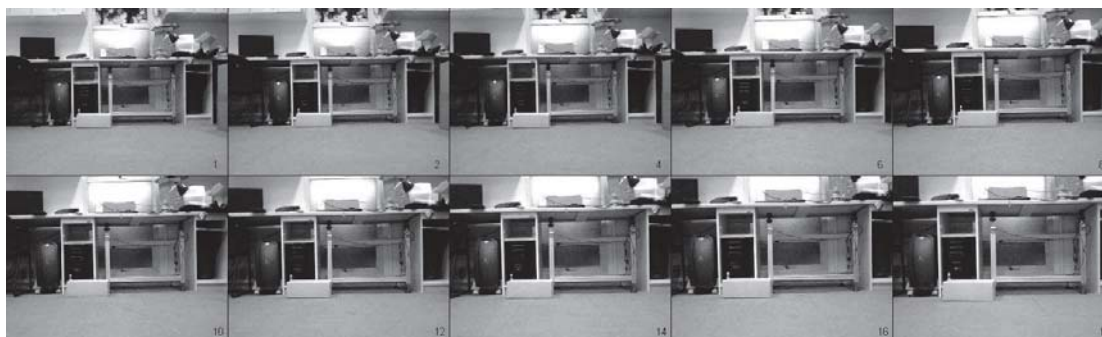


Figura 5-11 Imagens da câmera esquerda no deslocamento linear

A simulação inicial no ambiente 3D, apresentou baixa a sensibilidade do algoritmo de estimação da odometria, devido à correlação incorreta de pontos nas diferentes imagens, o que exigiu a implementação de uma etapa para descartar correlações incorretas através do filtro descrito na seção 5.3.1. Nos experimentos práticos se fez o mesmo trajeto com duas intensidades de filtragem, de 30% e 70% .

A reconstrução por odometria visual da linha 1D, com o filtro em 30% e representada Figura 5-12 . A linha de cor azul representa a odometria visual e a linha de cor vermelha a odometria dos ecoders. As duas estimações são analisadas e comparadas, para se obter uma medida do erro.

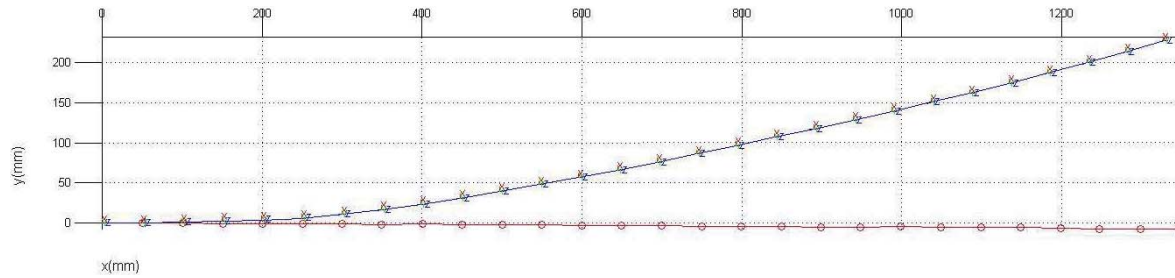


Figura 5-12 Reconstrução da odometria visual e dos encoders, trajeto linha 1D, filtro 30%

Na Figura 5-13 abaixo, apresentam-se os erros de orientação (pontos verdes) e de translação (pontos vermelhos). O erro de orientação local é cumulativo, diferente do erro de translação que é reinicializado em cada interação.

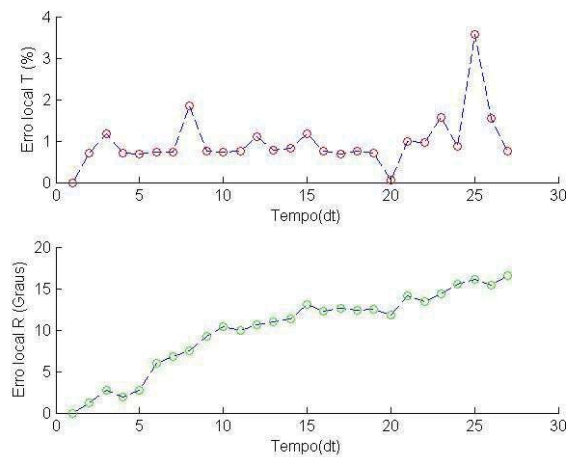


Figura 5-13 Erros locais da trajetória, linha 1D, com o filtro em 30%

As análises dos erros foram efetuadas através dos gráficos apresentados na Figura 5-14. Na parte superior, as barras verdes representam o erro cumulativo geral, que, foi estimado através da diferença da trajetória estimada e a referência dos encoders. As barras amarelas representam o número de *caraterísticas*, em relação à interação com maior número de pontos encontrados. O erro médio do trajeto é de 7,5%.

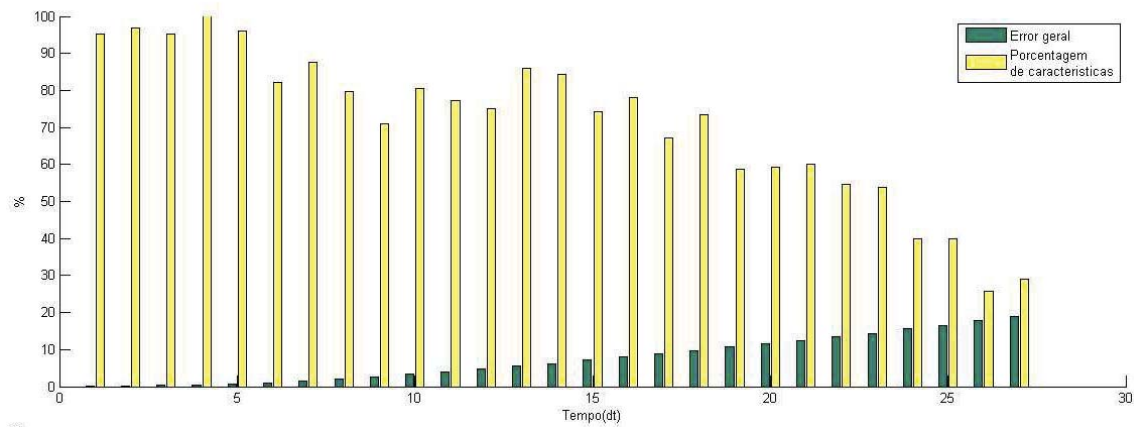


Figura 5-14 Erro e número de pontos da trajetória, distribuição normal e erro médio do trajeto linha 1D, filtro 30%.

O experimento anterior foi repetido mudando a configuração do filtro, ou seja, a intensidade que era de 30% mudou para 70%. A trajetória reconstruída e a de referência são apresentadas na Figura 5-15, representadas pelas linhas azuis e vermelhas respectivamente. A trajetória por visão apresenta um erro acumulativo, mas tem uma mudança de orientação na interdição 23-24 ($1100_{mm}Z$).

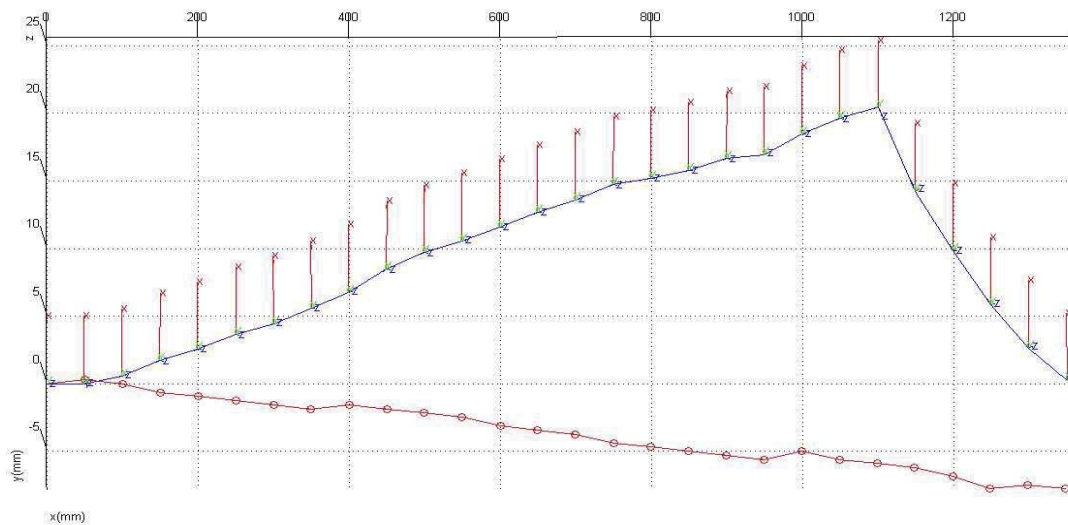


Figura 5-15 Reconstrução da odometria visual e dos encoders, trajeto linha 1D, filtro 70%.

Os erros locais são apresentados na Figura 5-16 abaixo. Erros locais de orientação são representados através dos pontos verdes e os de translação através dos pontos vermelhos. O erro local de rotação apresenta um erro de 6 graus na interdição [23-24], explicando a mudança de trajetória na Figura 5-15 anterior. O descarte das correlações erradas garante que o erro de orientação não seja cumulativo.

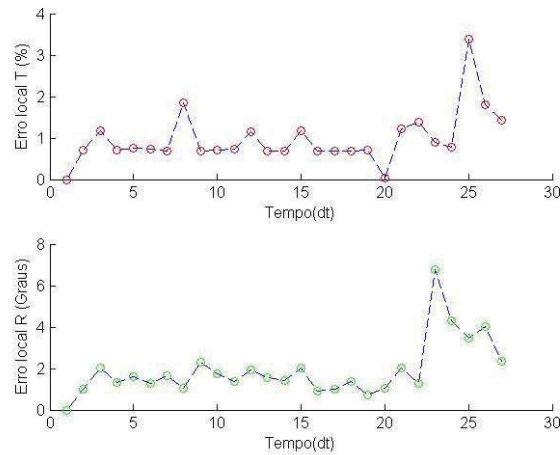


Figura 5-16 Erros locais da trajetória, linha 1D, com o filtro em 70%

Na Figura 5-17 o erro médio do trajeto é de 1,1% e o numero de correlações nas imagens decresceu devido as alterações feitas no filtro. O erro continua cumulativo, além de ser baixo.

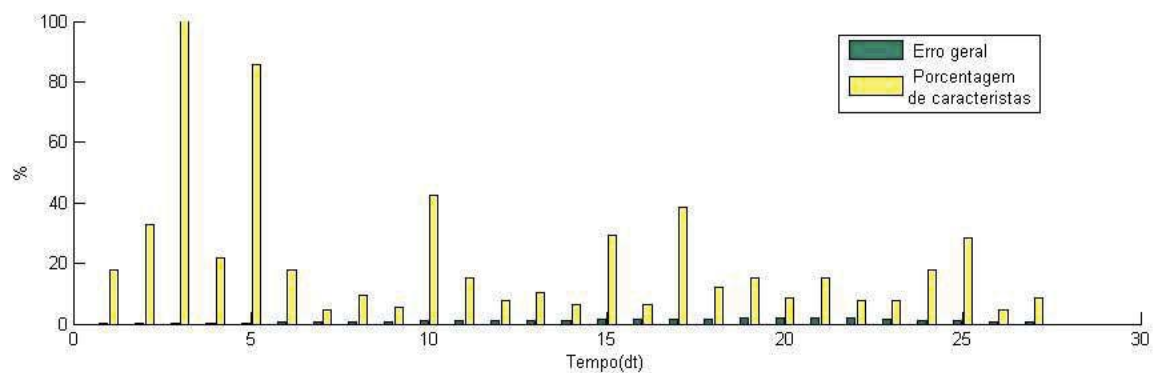


Figura 5-17 Erro e número de pontos da trajetória, distribuição normal e erro médio do trajeto linha 1D, filtro 70%.

5.3.3 Trajeto senoidal

O trajeto feito neste experimento é em formato senoidal (Figura 6-20 abaixo), desenhado para testar o algoritmo em trajetos com rotações. O trajeto foi discretizado, e cada intervalo é localizado com referência ao ponto de partida, utilizando a informação dos encoders e das câmeras.

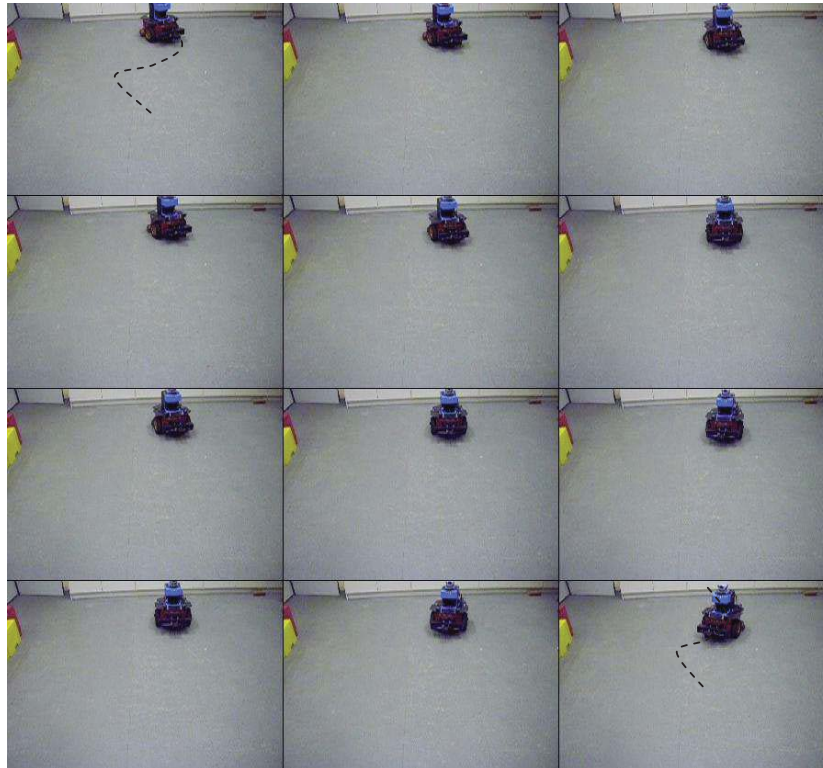


Figura 5-18 Imagens da navegação no trajeto senoidal

Na primeira parte do experimento o filtro é configurado com uma intensidade de 30% . Na Figura 5-19, são apresentados os resultados do trajeto. A linha vermelha representa a pose estimada pelos encoders, e a linha azul, representa a odometria reconstruída por visão. As duas estimações são analisadas e comparadas, para fornecer uma medida do erro.

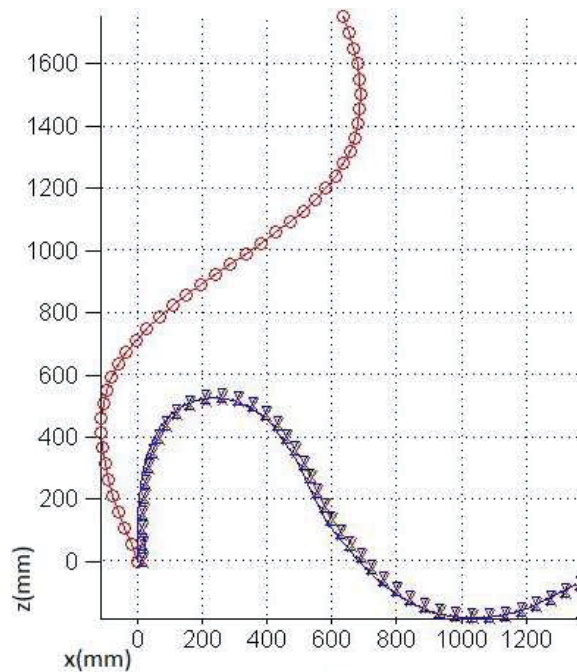


Figura 5-19 Reconstrução da odometria visual e dos encoders, trajeto senoidal 2D, filtro 30%.

O erro nos resultados da odometria visual é estimado em duas partes. Na primeira é feita uma estimativa local (em cada interação), utilizando como referência os dados dos encoders, no mesmo intervalo. Os erros nos vetores de deslocamento são estimados por diferença vetorial. Na segunda parte, é estimado o erro do trajeto reconstruído. A soma dos vetores de deslocamento (odometria) em cada intervalo que compõem o trajeto final gera um erro cumulativo. A Figura 5-20 abaixo apresenta os erros de orientação (pontos verdes) e de translação (pontos vermelhos). A translação não apresenta acumulação de erro, já que em cada deslocamento inicia-se do zero, diferente do erro de orientação, que compara a orientação dos vetores de deslocamento e se acumula de interação a interação.

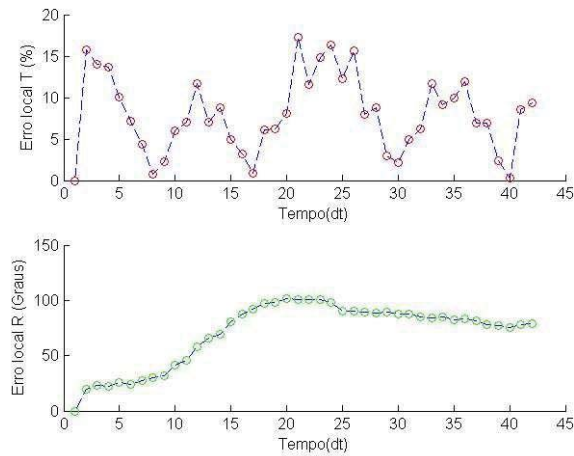


Figura 5-20 Erros locais da trajetória, senoidal 2D, com o filtro em 30%

O número de pontos correlatos e o erro acumulativo no trajeto são ilustrados na figura abaixo. As barras verdes representam o erro cumulativo geral, que foi estimado através da diferença entre a trajetória estimada e a referência. As barras amarelas representam os números de correlações (2.3.4) encontradas. Os valores são dados proporcionalmente, em relação ao maior número encontrado correlações, representado por 100%. Na parte inferior da Figura 5-21, é apresentada a distribuição normal e o erro médio. No trajeto deste experimento, com a configuração do filtro em 30%, o erro médio foi de 57%.

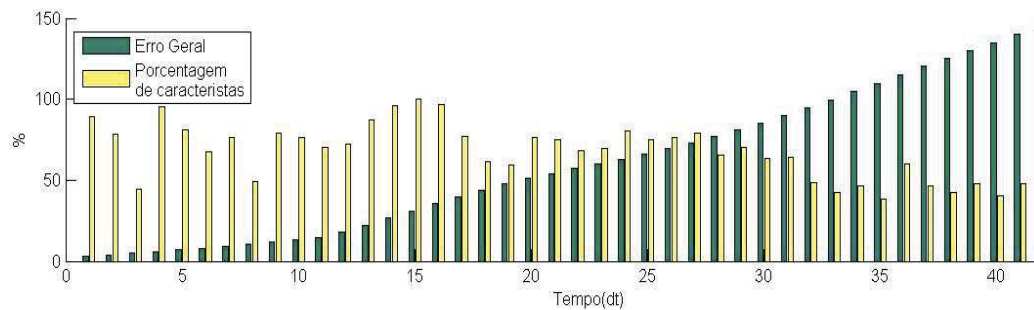


Figura 5-21 61 Erro e número de pontos da trajetória, distribuição normal e erro médio do trajeto , filtro 30%

Para ilustrar o comportamento do filtro, foram desenhadas as componentes vetoriais do movimento dos pontos correlatos em uma interdição, Figura 5-22 abaixo. As linhas vermelhas representam o comportamento de cada componente no trajeto. Os pontos azuis apresentam a média, que, quando são descartados pelo filtro, ficam em zero. Os pontos descartados são os que

têm maiores diferenças de comprimento ρ , com relação à média. O parâmetro de corte do filtro foi configurado para permitir diferenças até 70%, ou seja, uma rejeição de 30%.

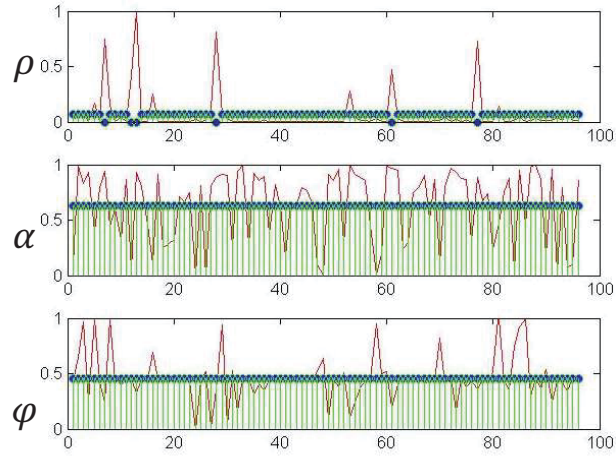


Figura 5-22 Correlações representadas como vetores em coordenadas esféricas

As maiores diferenças em ρ , representadas nos picos do gráfico anterior, foram ilustradas sobre as imagens somadas em dois instantes diferentes de tempo. A Figura 5-23 abaixo, contém as correlações de pontos no tempo anterior t e posterior $(t + 1)$ representado por uma cruz verde e um círculo vermelho respectivamente. As correlações são representadas pelos vetores amarelos que interligam os pontos.

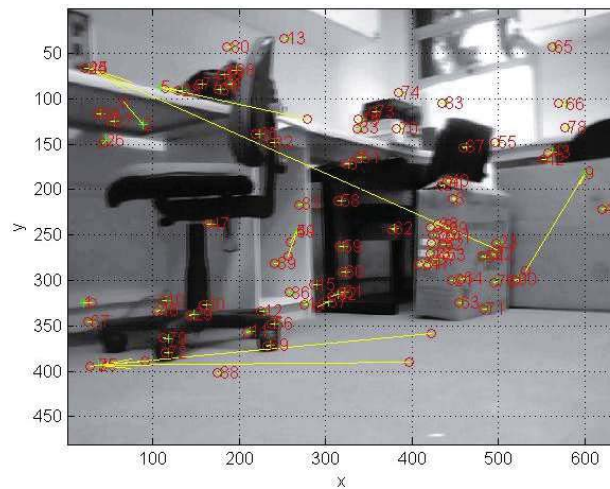


Figura 5-23 Correlações representadas como vetores sobre a somatória das imagens filtro 30%

Num segundo experimento, foi feito o mesmo trajeto senoidal, mas, com a intensidade do filtro em 70 %, garantido a qualidade dos pontos. Na Figura 5-24 abaixo representa a odometria, de referência (encoders) e a estimação por visão, linha vermelha e azul, respectivamente.

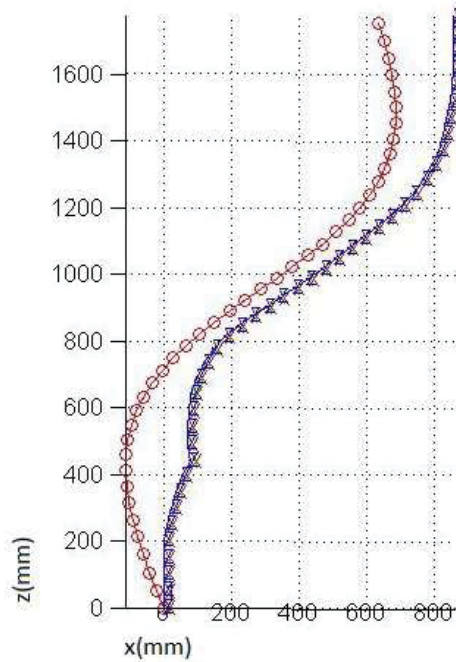


Figura 5-24 Reconstrução de uma trajetória em formato de senoidal 2D, com o filtro em 70%

O filtro removeu os pontos das correlações erradas apresentadas na Figura 5-23. O novo conjunto de correlações é apresentado abaixo, representado uma melhor qualidade nos pontos. Na Figura 5-25 as imagens de posição do robô em dois tempos consecutivos são apresentadas de forma superposta, como também, a representação por uma cruz verde e círculo vermelho dos pontos em cada interação.

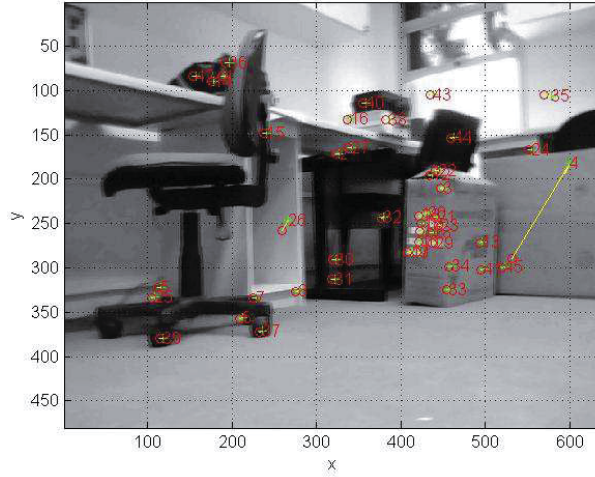


Figura 5-25 Correlações representadas como vetores sobre a somatória das imagens filtro 70%

O erro local foi calculado para o novo conjunto de pontos e apresentado na Figura 5-26 abaixo. A linha com os pontos vermelhos representam os erros locais de translação, e os pontos verdes os erros locais de rotação.

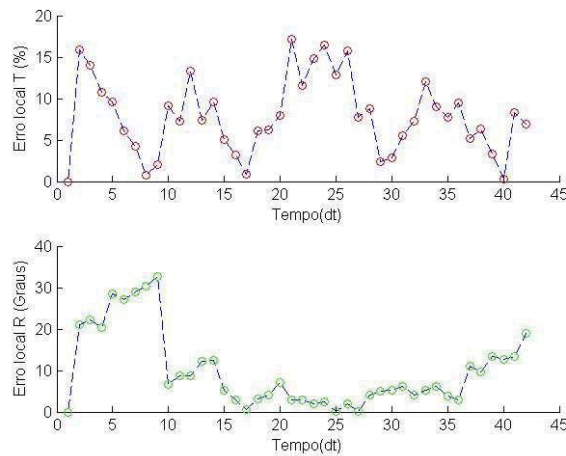
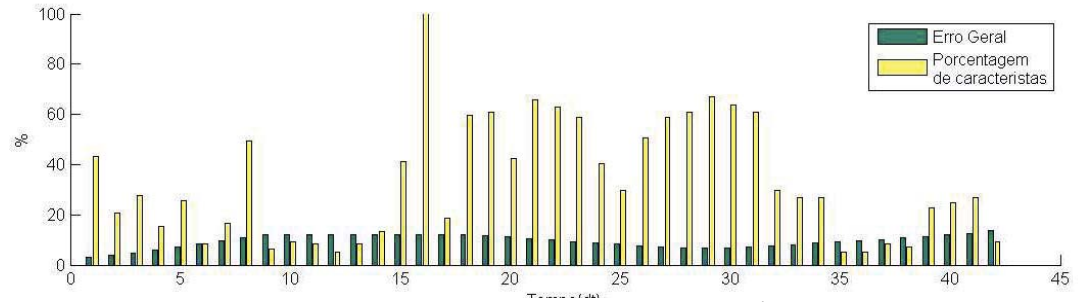


Figura 5-26 Erros locais da trajetória, senoidal 2D, com o filtro em 70%

O erro e a qualidade dos pontos encontrados com a nova configuração do filtro são apresentados no gráfico abaixo. As barras verdes representam o erro cumulativo geral e as barras amarelas o número de características, o erro é também incremental, mas muito menor. No

experimento com o filtro em 30% o erro foi de 57%, e com o filtro em 70% o erro médio na Figura 5-27 é de 9.5%.



**Figura 5-27 Erro e numero de pontos da trajetória,
distribuição normal e erro médio do trajeto senoidal 2D, filtro 70%**

6 DISCUSSÕES, CONCLUSÕES

Este capítulo apresenta a discussão dos resultados obtidos neste trabalho. Os interrelacionamentos dos erros e generalizações dos resultados são comparados com a literatura. Em seguida é feita uma síntese de tópicos e sugestões para os próximos estudos e trabalhos. Apresentam-se ainda as conclusões do trabalho.

O erro da trajetória estimada é influenciado pela intensidade do filtro e pela trajetória. Os resultados experimentais de navegação são apresentados na Tabela 2.

Trajetória	Intensidade Do filtro %	Erro local Translação	Erro local Rotação	Erro geral
Linha 1D	30	0-4%	0-18%	7,5%
	70	0-4%	0-8%	1,1%
Senoidal 2D	30	0-16%	0-110%	57%
	70	0-16%	0-31%	9,5%

Tabela 2

O aumento da intensidade do filtro de correlações de 30% para 70% resultou na redução do erro nas duas trajetórias. Na trajetória linear decresceu 6.4% e na senoidal 47.5%.

As trajetórias com rotações apresentam maior erro quando comparadas com as trajetórias lineares. O erro médio da trajetória linear experimental foi de 1.1%. Por outro lado, a trajetória senoidal apresenta um erro médio de 9,5%. Estes resultados estão dentro dos valores esperados por Steder no trabalho descrito na referência (Steder, et al., 2008), onde erros em trajetórias sem rotações são tipicamente entre 3.6% e 5.2% ,e entre 7.3% e 8.0% com rotações. Também estão próximos ao trabalho de (Stephen Se, 2001), onde o erro em trajetórias sem rotações foi de 1,12%, e com uma rotação de 5° de 5,83%.

A identificação de características correspondentes em duas imagens é possível com o uso do algoritmo SIFT. Embora houvesse robustez quanto a mudanças de rotações, escala e

iluminação, foi necessária a implementação de um filtro para remover os pontos com baixa qualidade de correlação.

O algoritmo de odometria visual foi testado num ambiente de interiores, composto por características estruturais que favorecem a correspondência de pontos. A análise em ambientes exteriores apresenta uma maior complexidade. No presente trabalho, os experimentos não foram feitos em ambientes externos, restringindo o campo de aplicação. Uma sugestão para os seguintes trabalhos é o teste do algoritmo em ambientes externos.

A velocidade de processamento do algoritmo de odometria visual, programado em Matlab e executado em um processador de 2,31 GHz, foi de 0,1f/s (quadros/segundo). As aplicações em tempo real requerem uma taxa de processamento de 60f/s. As sugestões para os seguintes trabalhos são a programação do algoritmo em linguagens de alto desempenho, como c++ e a execução em dispositivos de processamento paralelo, como GPU's ou MPI's.

O presente trabalho tem aplicações na indústria automobilística e nos robôs autônomos. Aplicações na indústria automotiva utilizam diversos sensores, como ultra-som nos pára-choques (Toyota, 2004),(Yasushi, 2006),(BMW), ou câmeras traseiras, que fornecem uma realimentação visual sobre a presença de obstáculos (Audi). O tratamento destas imagens pode fornecer maiores informações ao usuário. Por outro lado, grupos de pesquisa trabalham na criação de sistemas de navegação autônomos, capazes de conduzir um veículo automotivo de um ponto a outro sem a intervenção humana, onde a informação de odometria é indispensável.

Referências

1. M., R. SMITH, E P. Estimating Uncertain Spatial Relationships in Robotics. *Auton. Robot Veh.*, vol. 8, pp. 167–193,, Vol. 8, pp. 167–193, 1990.
2. SILVEIRA, G., MALIS, E. E RIVES, An Efficient Direct Approach to Visual SLAM. P. 5, 2008, #**IEEE_J_RO**#, Vol. 24, pp. 969-979.
3. 3.SILVEIRA, G., MALIS, E. E RIVES, An Efficient Direct Method for Improving visual SLAM. P. 2007. pp. 4090-4095.
4. DAVISON, A. J. E MURRAY, D. W. Simultaneous localization and map-building using active vision. #**IEEE_J_PAMI**#, Vol. 24, pp. 865-880,7, 2002.
5. ZHU, ZHIGANG, ET AL. Keeping smart, omnidirectional eyes on you [adaptive panoramic stereovision]. *Ieee Robotics \& Automation Magazine*, Vol. 11, pp. 69-78, 4, 2004.
6. SAEEDI, P., LAWRENCE, P. D. E LOWE, D. G.Vision-based 3-D trajectory tracking for unknown environments. #**IEEE_J_RO**#, Vol. 22, pp. 119-136,1, 2006.
7. STEDER, B., ET AL. Visual SLAM for Flying Vehicles. #**IEEE_J_RO**#, Vol. 24, pp. 1088-1093, 5, 2008.
8. PAZ, L. M., ET AL. Large-Scale 6-DOF SLAM With Stereo-in-Hand. #**IEEE_J_RO**#, Vol. 5, 2008. 24, pp. 946-957.
9. MAHON, I., ET AL. Efficient View-Based SLAM Using Visual Loop Closures. #**IEEE_J_RO**#, 5, 2008, Vol. 24, pp. 1002-1014.

10. LOWE, DAVID G. VANCOUVER, B.C. Distinctive Image Features from Scale-Invariant Keypoints.: **International Journal of Computer Vision**, 2004.
11. STEPHENS, C.G. HARRIS E M. A Combined Corner and Edge. **Proc. Fourth Alvey Vision Conf.** 1988. 147-151, s.l.
12. BOUGUET, J. Y. **Camera calibration toolbox for Matlab**. 2003.
13. KURKA, P.R DELGADO V.J., MINGOTO J.C., ROJAS O. Automatic Estimation of Camera Parameters from a Solid Calibration Box, **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, Aceito para publicação, em abril de 2012. ISSN: 1678-5878 .
14. EMANUELE TRUCCO, ALESSANDRO VERRI. **Introductory Techniques for 3-D Computer Vision**. s.l. : Prentice Hall, 1998.
15. TOYOTA. **Toyota Ultrasonic system bumpers** . [Online].
<http://www.toyota.com.au/accessories/exterior-accessories/parkassist> ParkAssist. 2004
16. YASUSHI, T. YU, E. TOMOHIKO, AND K. YUKIKO. Development of intelligent parking assist. Japan : **journal of the Society of Automotive Engineers of Japan**, 2006, Vol. 60.
17. BMW. **Park Distance Control**. [Online] [Citado em: 15 de 05 de 2011.]
http://www.bmw.com/com/en/insights/technology/technology_guide/articles/park_distance_control.html?source=categories&article=park_distance_control.
18. AUDI. **Automatica parking system** . [Online] [Citado em: 15 de 05 de 2011.]
http://www.audiusa.com/us/brand/en/tools/advice/glossary/audi_parking_system0.browse.r.carline_s6limo.html.

19. LIU, KANG-ZHI, DAO, MINH QUAN E INOUE, TAKUYA. Theory and experiments on automatic parking systems. **Control, automation, robotics and vision conference, 2004. Icarcv 2004 8th**. 2004. Vol. 2, pp. 861-866.

20. ROLAND, SIEGWART E NOURBAKHSI, ILHA R. **Introduction to autonomous Mobile Robots**. s.l. : The MIT Press, 2004. ISBN 0-262-19502.

21. FINTZEL, K., ET AL. 3D parking assistant system. **Intelligent Vehicles Symposium, 2004 IEEE**. 2004. pp. 881-886.

22. PARK, SOON YOUNG, ET AL. Monocular-vision-based parking assistance for constrained environment. **Intelligent Transportation Systems (ITSC) 13th International IEEE Conference on**. 2010. pp. 617-622.

23. AGARWAL, V., MURALI, N. V. E CHANDRAMOULI, C. A Cost-Effective Ultrasonic Sensor-Based Driver-Assistance System for Congested Traffic Conditions., **#IEEE_J_ITS#**, 3, 2009, Vol. 10, pp. 486-498.

24. CHANG, TANG-HSIEN E CHOU, CHEN-JU. Rear-end collision warning system on account of a rear-end monitoring camera. **Intelligent Vehicles Symposium, 2009 IEEE**. 2009. pp. 913-917.

25. KWON, GOO-RAK, ET AL. Automatic guideline generation for parking assistance system based on on-screen display. **Consumer Electronics, 2009. ISCE '09. IEEE 13th International Symposium**. 2009. pp. 845-846.

26. LI, S. E HAI, Y. Easy Calibration of a Blind-Spot-Free Fisheye Camera System Using a Scene of a Parking Space **#IEEE_J_ITS#**, 1, 2011, Vol. 12, pp. 232-242.

27. LI, T.-H. S. E CHANG, SHIH-JIE. Autonomous fuzzy parking control of a car-like mobile robot. 4, 2003, **IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans**, Vol. 33, pp. 451-465.

28. JEEVAN, P., ET AL. Realizing Autonomous Valet Parking with automotive grade sensors. **Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on**. 2010. pp. 3824-3829.

29. HSU, TSUNG HUA, ET AL. Development of an automatic parking system for vehicle. **Vehicle Power and Propulsion Conference, 2008. VPPC '08. IEEE**. 2008. pp. 1-6.

30. CHOU, LI-DER, SHEU, CHUN-CHENG E CHEN, HOME-WAY. Design and Prototype Implementation of A Novel Automatic Vehicle Parking System. **Hybrid Information Technology, 2006. ICHIT '06. International Conference on..** 2006. Vol. 2, pp. 292-297.

31. SEETHARAMAN, G. LAKHOTIA, A. E BLASCH, E. P. Unmanned Vehicles Come of Age: The DARPA Grand Challenge. **Computer**, 12, 2006, , Vol. 39, pp. 26-29.

32. SONKA, MILAN. **Image processing, analysis, and machine vision**. S.l. : Thomson, 2008.

33. ESPINOSA, CARLOS ANDRÉS DIAZ. **Uma Aplicação de Navegação Robótica autônoma através de Visão Computacional Estéreo**. Campinas : Unicamp, 2010.

34. DELGADO, V.J., KURKA, P.R., CARDOZO, E. Visual odometry in mobile robots. **2011 IEEE IX Latin American and IEEE Colombian Conference on Automatic Control and Industry Applications (LARC)**, 2011 , pp 1 – 4.

35. Stephen Se, David Lowe, Jim Little. Vision-based Mobile Robot Localization And Mapping using Scale-Invariant Features [Periódico]. - Canada : **International Conference on Robotics & Automation Ieee, 2001.** - Vols. Seoul, Korea . May 21-26, 2001.
36. A. Vedaldi, B. Fulkerson. VLFeat - An open and portable library of computer vision algorithms Vedaldi. **Proceedings of the 18th annual ACM international conference on Multimedia**, 2010,

ANEXOS A - Produção Bibliográfica

Visual Odometry In Mobile Robots

Delgado, v.j., Kurka, p.r., Cardozo,

Abstract— The paper presents an application of visual odometry, through reconstruction of the path of a mobile robot, using a stereoscopic camera system. The scale invariant feature transformation algorithm (SIFT), is used to process the images and locate keypoints in a 3D euclidian coordinates space. The path of a Pioneer mobile robot is estimated using the proposed technique. (Delgado, V.J, 2011)

Automatic Estimation of Camera Parameters from a Solid Calibration Box

Kurka, P.R.G.; Delgado, J.V.; Mingoto, C.R.; Rojas, O.E.R.

Abstract— The paper presents a simple and automatic procedure to estimate the intrinsic and extrinsic parameters of a camera, based on the image of a standard calibration box. Image processing techniques are used to determine the relevant image points to be used in the calibration procedure. A non-recursive solution scheme is proposed to estimate the intrinsic and extrinsic camera parameters. Simulated and real applications are presented to illustrate the use and performance of the proposed technique. (Kurka, et al., 2012)

ANEXOS A - Diagrama de fluxo do algoritmo de odometria

